ULTRIX

Guide to System and Network Setup

Order Number: AA-ME88B-TE
June 1990

Product Version: ULTRIX Version 4.0 or higher

This guide identifies a series of tasks that are basic to setting up a system, establishing it on a network, and setting up a distributed environment. For each task that it identifies, the Guide provides step-by-step information on how to complete it and pointers to more information on the topic. If the ULTRIX software has a script that simplifies completing the task, the script is described.

digital equipment corporation
maynard, massachusetts

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Digital or its affiliated companies.

The following are trademarks of Digital Equipment Corporation:

digital
CDA
DDIF
DDIS
DEC
DECnet
DECstation
DECUS
DECwindows
DTIF
MASSBUS
MicroVAX
Q-bus
ULTRIX
ULTRIX Mail Connection
ULTRIX Worksystem Software
UNIBUS
VAX
VAXstation
VMS
VMS/ULTRIX Connection
VT
XUI

Ethernet is a registered trademark of Xerox Corporation.

UNIX is a registered trademark of AT&T in the USA and other countries.

Contents

Preface

1 System Setup

2 Network Setup

3 Distributed System Services Setup

Preface

The *Guide to System and Network Setup* identifies a series of tasks that are basic to setting up your system and establishing it on a network, and provides step-by-step information on how to complete each task. If the ULTRIX software has a script that simplifies completing the task, the script is described. The Guide also tells you where you can find additional information on each topic.

Prerequisites

This book presumes that you have successfully completed your ULTRIX installation, and that you correctly installed any optional subsets that you need. The basic installation provides the following subsets: Base System, Kernel Configuration Files, TCP/IP Networking Utilities, Network File System Utilities, Extended (Berkeley) Mailer, X11/DECwindows 75dpi Fonts, X11/DECwindows Servers, and X11/DECwindows User Environment.

You must add the optional software subset, Printer Support Environment, to your system if you intend for your system to access printers either locally or remotely. See the *Advanced Installation Guide* for more information on the mandatory and optional subsets provided by the ULTRIX software installation.

Scope

This book addresses first-time system and network setup on a timesharing system with multiple users. It does not address the following areas:

- System security—For information on setting up a secure local host or a secure networked environment, see the *Security Guide for Administrators* or the *Guide to Kerberos*.
- Diskless client setup—For information on managing the diskless client environment, see the *Guide to Diskless Management Services*.

Although the Guide assumes a first-time installation, most of the task descriptions and steps are applicable if you are modifying an existing environment. If you cannot complete a particular task because of your system or network configuration, the software indicates that there is a conflict and what to do resolve it.

Audience

The audience for this document is a novice system or network administrator. You should be familiar with ULTRIX commands and utilities, but you do not need extensive computer experience.

Organization

The *Guide to System and Network Setup* is divided into three parts:

- Part 1: System Setup

  This part outlines the system management tasks required to set up your local system.

- Part 2: Network Setup

  This part outlines the tasks required to establish a TCP/IP local area network, and to access remote resources.

- Part 3: Distributed System Services Setup

  This part outlines the tasks required to establish a distributed environment.

Each task is organized in the following way:

- Before You Start—a list of the information that you should gather before attempting the task
- Steps—a description of how to complete the task
- Graphic—an illustration depicting the task
- See Also—pointers to other sources of information

Conventions

`%` The default user prompt is your system name followed by a right angle bracket. In this manual, a percent sign (%) is used to represent this prompt.

`#` A number sign is the default superuser prompt.

`user input` This bold typeface is used in interactive examples to indicate typed user input.

`system output` This typeface is used in interactive examples to indicate system output and also in code examples and other screen displays. In text, this typeface is used to indicate the exact name of a command, option, partition, pathname, directory, or file.

UPPERCASE
lowercase
The ULTRIX system differentiates between lowercase and uppercase characters. Literal strings that appear in text, examples, syntax descriptions, and function definitions must be typed exactly as shown.

rlogin In syntax descriptions and function definitions, this typeface is used to indicate terms that you must type exactly as shown.

macro In text, bold type is used to introduce new terms.

[ ] In syntax descriptions and function definitions, brackets indicate items that are optional.

`cat(1)`

Cross-references to the *ULTRIX Reference Pages* include the appropriate section number in parentheses. For example, a reference to `cat(1)` indicates that you can find the material on the `cat` command in Section 1 of the reference pages.

CTRL/*x*

This symbol is used in examples to indicate that you must hold down the CTRL key while pressing the key *x* that follows the slash. When you use this key combination, the system sometimes echoes the resulting character, using a circumflex (^) to represent the CTRL key (for example, ^C for CTRL/C). Sometimes the sequence is not echoed.

1 System Setup

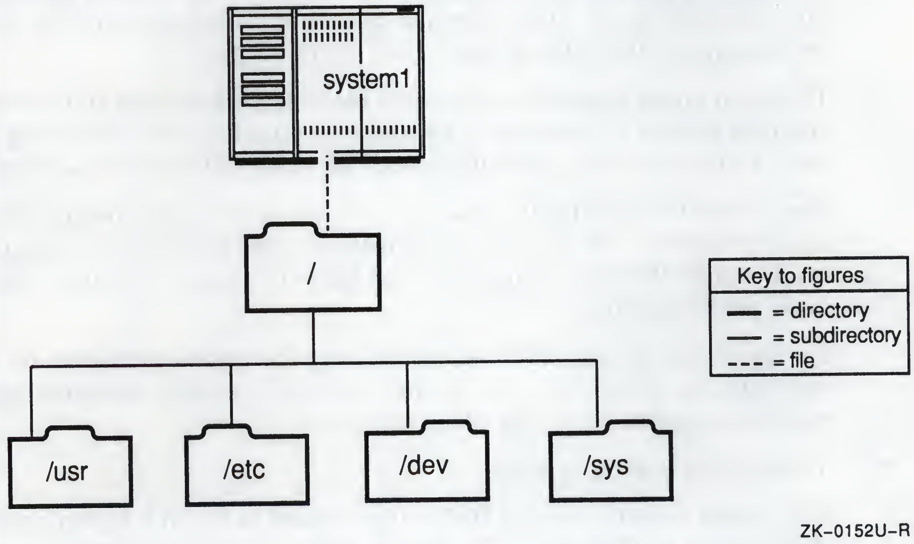
When setting up your local environment you must consider your system configuration, what system information you want to track, how much disk space you have, what devices are attached, and whether your machine will be part of a network.

This section describes a series of generic system setup tasks that will help you get your system up and running. For general information on system setup, see the *Guide to System Environment Setup*.

The following tasks are described in this section:

- Adding users with `adduser`
- Adding pseudoterminal devices
- Adding local area transport (LAT) devices
- Adding printers with `lprsetup`
- Establishing disk quotas

A variation on the following figure appears for each task. The figure depicts the system files created or changed in completing the task, as well as their directory tree structure. The key is applicable to all figures throughout this section:



Adding Users with adduser

The `adduser` command automates adding user accounts to the local system. It prompts you for information about the new user and then either creates the appropriate files, or adds the information you provide to existing system files.

Note

> The `adduser` command only adds users to the local system. If you are planning a distributed environment, see Part 3 for information on adding users.

For each new user account, the `adduser` command creates a home directory with generic startup files copied from `/usr/skel`, and a `bin` subdirectory.

You can also add users to group entries in the `/etc/group` file with the `adduser` command.

Before You Start

Before running the `adduser` command, you should gather the following information:

- The new user's login name, user identification number or UID (if the user has an account on another system in your environment), full name, login group, parent directory, and any other groups that should include the new user

  The system uses the UID, not the login name, to determine the identity of a user. Therefore, the UID of a particular user should be the same on all systems in a networked environment. If you are in a networked environment, and the user you are adding has an account on another system, specify the same UID on this system as on the other. See the `passwd`(5) reference page for information on interpreting the fields in the `/etc/passwd` file.

  The login group determines the group identification number (GID) for processes and files created by the user. If permissions on a file are set allowing group read, write or execute, other users with the same GID can access those files.

  The parent directory for the new user is the directory that contains the user's home directory. The user's home directory is the directory that the user logs in to. Users in different login groups can have their home directories under the same parent directory.

  See the *Guide to System Environment Setup* for more information on user accounts, and the `/etc/group` file. See the `chmod`(1) reference page for more information on setting file permissions.

- Your system's security level

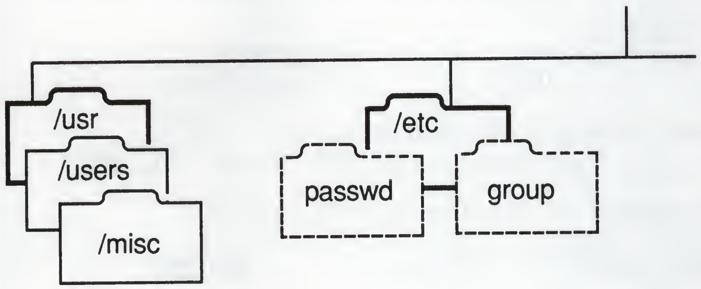
  The default security level is BSD. If you want to run at a higher security level (UPGRADE or ENHANCED), special system setup is required.

  If your system is running at the UPGRADE or ENHANCED level, the `adduser` command asks security-related questions that it does not ask if your system is running at the BSD level.

  See the *Security Guide for Administrators* for information about system setup and adding user accounts on more secure systems.

Steps

The following figure depicts the system files that are created or changed when you are running the adduser command:



You must be logged in as superuser to run the adduser command.

Note

> To terminate adduser with no modifications to system files press CTRL/C.

1. Type the following:

```
# adduser
```

2. Enter the login name of the new user, the UID, the full name, and the login group.

   The login name must be less than 9 characters long, and can not contain colons.

   If the new user has an account on another system in your environment, set the UID to match that of the account on the other system. You can also specify a particular UID, if you have a numbering scheme for users in your environment. Otherwise, accept the default.

   The following example shows how to add a new user named John A. Laker, with a login name of jal, to the staff3 login group. In this example, the default UID is accepted:

```
Enter login name for new user (initials, first or last name):  jal
Enter uid for new user [268]:
Enter full name for new user:   John A. Laker
What login group should this user go into [ users ] ?  staff3
```

   The default login group is users, but you can specify any group as a login group. The group name cannot contain colons. If you specify a group that does not exist, the adduser command indicates that the group is unknown, and then asks if you want to add it to the /etc/group file. If you choose to add the new group to the /etc/group file, the adduser command prompts you for a number for the new group. Either accept the default group number by pressing the RETURN key or specify another number.

The following example shows how to add the group staff3 to the /etc/group file, and assign it the group number 79:

```
Unknown group: staff3. Known groups are:

system          daemon          uucp            rsrv3
bin             tty             kmem            authread
news            rsrv9           staff           ris
users           guest           operator        ingres

Do you want to add group staff3 to the
                /etc/group file? [yes]: RETURN

Adding new group to /etc/group file...

Enter group number for new group [79]: RETURN
```

3. Indicate other groups that should include the new user.

   If you specify another group that does not exist, the adduser command goes through the sequence described in step 2.

4. Specify the parent directory for the new user.

   The default parent directory is /usr/users. To accept the default, press the RETURN key. If you want to specify a different parent directory, enter the directory pathname when the adduser command prompts you for it. If the parent directory you specify does not exist, the adduser command asks if you want to create it. The new user's home directory is automatically created as a subdirectory of the parent directory.

   The following example shows how to specify /usr/staff/research as the new user jal's parent directory:

```
Enter parent directory for jal [/usr/users]:  /usr/staff/research

/usr/staff/research not found,
                do you want to create it? [yes]: RETURN
```

5. Select a login shell for the new user.

   The adduser command displays a list of the supported login shells. The default shell is /bin/csh. Press the RETURN key to accept the default, or enter another shell.

```
The shells are:

/bin/sh         /bin/csh        /usr/bin/ksh    /usr/bin/sh5

Enter the users login shell name [/bin/csh]: RETURN
```

   Normally, the login shell you select appears as a field in the new user's entry in the /etc/passwd file. However, if you select the Bourne shell, /bin/sh, as the login shell, the field that designates the login shell is left blank.

Note

If you enter a shell other than one that is listed, unprivileged users cannot change their shell.

6. Enter and verify a password for the new user.

   Each user account added with the adduser command must have a password that is at least six characters long associated with it. The adduser command displays the following information, and then prompts you to enter and verify a password for the new user:

   ```
   Adding new user ...
   Creating home directory...
   Until the password is set for jal they will not be able to login.
   Enter new password:
   Verify:
   ```

   After you have entered and verified the password, adduser exits.

See Also

chmod(1), passwd(1), group(5), adduser(8), removeuser(8), vipw(8)

Guide to System Environment Setup
Security Guide for Administrators

Adding Pseudoterminal Devices

Pseudoterminals enable users to access a system using the network. A pseudoterminal is a pair of character devices that emulates a hardware terminal connection to the system. Instead of hardware, however, there is a master device (`/dev/pty`*xx*) and a slave device (`/dev/tty`*xx*).

The following processes use pseudoterminal lines: `xterm`, `dxterm`, `script`, `rlogind`, `dlogind`, `telnetd`, and `dgatewayd`.

Pseudoterminal lines are created in sets of 16. The ULTRIX software provides a default of 32 pty lines, which corresponds to `pty0` and `pty1`.

For some installations, the default number of `pty` devices is adequate. However, as your user community grows, and each user wants to run multiple sessions on one or more timesharing machines in your environment, the machines may run out of available `pty` lines. The following error message is common, and results from too few `pty` devices being configured when a user tries establish a remote session using the `rlogin` command:

```
Insufficient network resources
```

Before You Start

Before adding pseudoterminal (or `pty`) lines to your system, you should gather the following information:

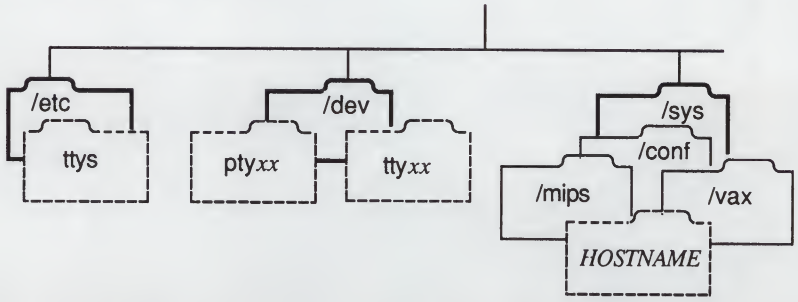
- Verify that `pty` is in the pseudo-device section of your system configuration file

  The pathname of the system configuration file is `/usr/sys/conf/vax` or `/usr/sys/conf/mips`, depending on whether yours is a VAX or RISC machine. The system configuration file name is is the host name of the machine in uppercase. Include the following line in the file if it is not already there:

  ```
  pseudo-device   pty
  ```

Steps

The following figure depicts the system files that are created or changed when you add pty lines to your system:



You must be logged in as superuser to complete the following steps.

1. Edit the system configuration file.

   Enter the number of lines you want your system to support next to the `pty` entry in the pseudo-device section. The number, which should be a multiple of

16, represents the total number of pseudoterminal lines that your system supports.

For example, if you want your system to support 48 pseudoterminal lines, edit the configuration file to read as follows:

```
pseudo-device   pty     48
```

You can configure a maximum of 176 lines.

2. Rebuild your kernel with the `doconfig` command and the `-c` option. The `-c` option specifies the use of the existing system configuration file to rebuild the kernel. The following command rebuilds the kernel of a system called `host1`:

   ```
   # doconfig -c HOST1
   ```

   See the `doconfig`(8) reference page for more information.

3. Go to the `/dev` directory and run the `MAKEDEV` command:

   ```
   # cd /dev
   # MAKEDEV pty2
   ```

4. Record the special file information that is displayed by `MAKEDEV`.

   The `MAKEDEV` display is similar to the following:

   ```
   MAKEDEV: special file(s) for pty2:
   ptyr0 ttyr0 ptyr1 ttyr1 ptyr2 ttyr2 ptyr3 ttyr3 ptyr4 ttyr4
   ptyr5 ttyr5 ptyr6 ttyr6 ptyr7 ttyr7 ptyr8 ttyr8 ptyr9 ttyr9
   ptyra ttyra ptyrb ttyrb ptyrc ttyrc ptyrd ttyrd ptyre ttyre
   ptyrf ttyrf
   ```

5. Edit the `/etc/ttys` file.

   Add the tty*xx* half of the pseudoterminal device pair to the `/etc/ttys` file. For example, edit the `/etc/ttys` file to read as follows:

   ```
   ttyr0   none          network
   ttyr1   none          network
   ttyr2   none          network
   ttyr3   none          network
   ttyr4   none          network
   ttyr5   none          network
   ttyr6   none          network
   ttyr7   none          network
   ttyr8   none          network
   ttyr9   none          network
   ttyra   none          network
   ttyrb   none          network
   ttyrc   none          network
   ttyrd   none          network
   ttyre   none          network
   ttyrf   none          network
   ```

   The first field indicates the name of the device special file. The second field indicates the command to be executed at startup. The third field is the type of terminal normally connected to the terminal special file.

See Also

`pty`(4), `ttys`(5), `doconfig`(8), `MAKEDEV`(8)

Guide to System Environment Setup

Adding Local Area Transport (LAT) Devices

The lta terminal driver provides support for remote terminals using the Local Area Transport (LAT) protocol. The LAT protocol allows users to access hosts on a local area network (LAN).

You can also set up printers to use the LAT to queue jobs. For information on setting up LAT printers, see the *Guide to Ethernet Communications Servers*.

LAT lines are created in sets of 16. The ULTRIX software provides a default of 16 LAT lines, which corresponds to lta0. For some installations, the default number of LAT devices is adequate. However, as your user community grows, and each user wants to run multiple sessions on one or more timesharing machine in your environment, the machines may run out of available LAT devices. When a user tries to connect using the LAT protocol to a timesharing machine that does not have enough LAT devices configured, the timesharing system returns the following error message:

```
Insufficient network resources
```

Before You Start

Before adding LAT lines to your system, you should gather the following information:

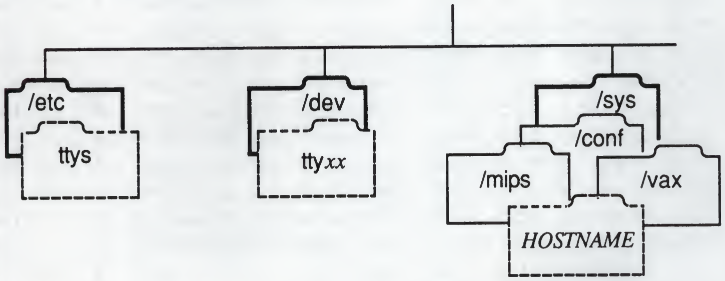
- Verify that LAT is in the options section of your system configuration file, and that lat and lta are in the pseudo-device section of your system configuration file

  The pathname of the system configuration file is /usr/sys/conf/vax or /usr/sys/conf/mips, depending on whether yours is a VAX or RISC machine. The system configuration file name is is the host name of the machine in uppercase. Include the following lines in the file, if they are not already there:

  ```
  options         LAT
        .
        .
        .
  pseudo-device   lat
  pseudo-device   lta
  ```

Steps

The following figure depicts the system files that are created or modified when you add LAT lines to your system:



If you are adding LAT devices to your system, complete all of the following steps. If the default number of LAT devices is adequate for your site, complete steps 6 and 7 only.

You must be logged in as superuser to complete the following steps.

1. Edit the system configuration file.

   Enter the number of lines you want your system to support next to the `lta` entry in the pseudo-device section. The number, which should be a multiple of 16, represents the total number of LAT lines that your system supports.

   For example, if you want your system to support 48 LAT devices, edit the configuration file to read as follows:

   ```
   pseudo-device   lta     48
   ```

   You can configure a maximum of 256 lines.

2. Rebuild your kernel with the `doconfig` command and the `-c` option. The `-c` option specifies the use of the existing system configuration file to rebuild the kernel. The following command rebuilds the kernel of a system called `host1`:

   ```
   # doconfig -c HOST1
   ```

   See the `doconfig`(8) reference page for more information.

3. Go to the `/dev` directory and run the `MAKEDEV` command:

   ```
   # cd /dev
   # MAKEDEV lta1
   ```

4. Record the special file information that is displayed by `MAKEDEV`. The actual special file names vary depending on how many other terminal devices are already configured.

   The `MAKEDEV` display is similar to the following:

   ```
   MAKEDEV: special file(s) for lta1:
   tty16 tty17 tty18 tty19 tty20 tty21 tty22 tty23
   tty24 tty25 tty26 tty27 tty28 tty29 tty30 tty31
   ```

5. Edit the /etc/ttys file.

   Add the tty*xx* displayed in the step above to the /etc/ttys file. For LAT devices, edit the /etc/ttys file in a similar manner to the following:

```
tty16  "/etc/getty std.9600" network on nomodem  # lat terminal
tty17  "/etc/getty std.9600" network on nomodem  # lat terminal
tty18  "/etc/getty std.9600" network on nomodem  # lat terminal
tty19  "/etc/getty std.9600" network on nomodem  # lat terminal
tty20  "/etc/getty std.9600" network on nomodem  # lat terminal
tty21  "/etc/getty std.9600" network on nomodem  # lat terminal
tty22  "/etc/getty std.9600" network on nomodem  # lat terminal
tty23  "/etc/getty std.9600" network on nomodem  # lat terminal
tty24  "/etc/getty std.9600" network on nomodem  # lat terminal
tty25  "/etc/getty std.9600" network on nomodem  # lat terminal
tty26  "/etc/getty std.9600" network on nomodem  # lat terminal
tty27  "/etc/getty std.9600" network on nomodem  # lat terminal
tty28  "/etc/getty std.9600" network on nomodem  # lat terminal
tty29  "/etc/getty std.9600" network on nomodem  # lat terminal
tty30  "/etc/getty std.9600" network on nomodem  # lat terminal
tty31  "/etc/getty std.9600" network on nomodem  # lat terminal
```

   The first field (tty*xx*) specifies the device special file name. The second field ("/etc/getty std.9600") specifies the command run by the /etc/getty command at login. The third field (network) specifies the terminal type used on this line. The fourth field (on) enables logins on this line. The fifth field (nomodem) specifies that modem signals on this line should be ignored. The sixth field (# lat terminal) is a comment indicating that this is a LAT line. See the ttys(5) reference page for more information.

6. Edit the /etc/rc.local file.

   Place an entry similar to the following after the sendmail entry in your /etc/rc.local file:

```
[ -f /etc/lcp ] && {
        /etc/lcp -s & echo -n ’ starting LAT’        > /dev/console
}
```

7. Reboot your system.

   Use the shutdown command with the -r option to reboot. The following command immediately performs an orderly shutdown and automatic reboot:

```
# /etc/shutdown -r now
```

See Also

lta(4), ttys(5), doconfig(8), lcp(8), MAKEDEV(8)

Guide to Ethernet Communications Servers
Guide to System Environment Setup

Adding Printers with lprsetup

The `lprsetup` command allows you to add local and remote printers and PrintServers to your system. The `lprsetup` command tailors the information it prompts you for depending on the type of printer you specify. With the information you provide, the `lprsetup` command then does the following:

- Creates an `/etc/printcap` entry
- Creates a spooling directory
- Creates accounting files
- Creates error log files
- Modifies the `/etc/ttys` file

Note

If you did not add the optional software subset ULTPRINT400 (for VAX machines) or UDTPRINT400 (for RISC machines) at installation, you must do so before attempting to use the `lprsetup` command. See the *Guide to System Environment Setup* and the `setld`(8) reference page for information on adding software subsets.

Before You Start

Before attempting to use the `lprsetup` command, you must have an understanding of the printer parameters that you are setting and the format of the `/etc/printcap` file. For an explanation of the printer parameters, see the *Guide to System Environment Setup* and the `printcap`(5) reference page. For sample `/etc/printcap` entries, see the `/etc/printcap.examples` file.

Before running the `lprsetup` command, you should gather the following information:

- The type of printer you are adding

  The `lprsetup` command provides default settings based on the type of printer you specify.

- The type of connection to your system and the appropriate values for the mandatory printer parameters

  Printer connections fall into the following categories:

  - Device—local connection to a serial or parallel port

    If the printer is directly connected to a serial or parallel port, you must define a device for it to open for its output (**`lp`** in the `/etc/printcap` file). The default that `lprsetup` provides is usually adequate.

- LAT—connection to a LAT port

  If the printer is connected to a LAT port, you must define a device for it to open for its output (**lp** in the `/etc/printcap` file), and an entry in the name field for LAT port characteristics (**op** in the `/etc/printcap` file). You must also enter the LAT terminal server node name (**ts** in the `/etc/printcap` file) and the default object service parameter (**os** in the `/etc/printcap` file). The default settings that `lprsetup` provides for **lp** and **os** are usually correct. No default settings are provided for **op** and **ts**.

  See the *Guide to Ethernet Communications Servers* for information on setting up a printer using a LAT terminal server.

- Remote—submits jobs to a remote machine

  If the printer is a remote printer you must specify the name of the machine from which it can be accessed (**rm** in the `/etc/printcap` file), and a name by which the remote machine knows the printer (**rp** in the `/etc/printcap` file). The **lp** parameter is left null by default. No default settings are provided for **rm** and **rp**.

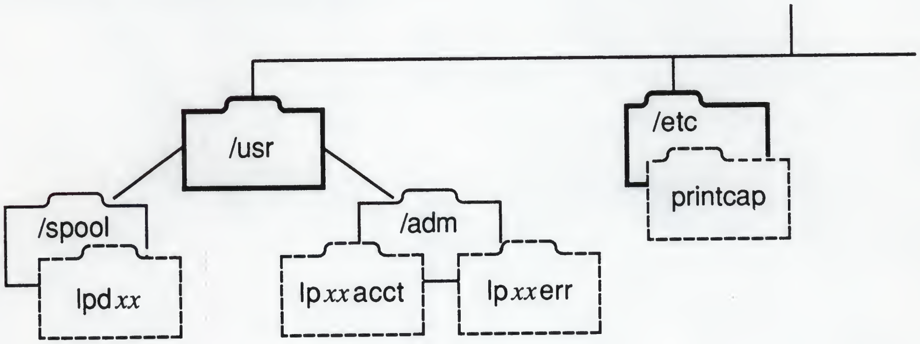
- Network—submits jobs to a PrintServer using the network

  For PrintServers you must define the appropriate output filter (**of** in the `/etc/printcap` file) and the PrintServer's node name. For PrintServers running TCP/IP the output filter is `of=/usr/lib/lpdfilters/iplpscomm`.

  The **Dl** parameter is automatically set to `Dl=/usr/lib/lpdfilters/lps_v3.a`, which is correct for Version 3.0 of the PrintServer supporting host software. If the PrintServer supporting host software is Version 2.0 or 2.1, you must set the **Dl** parameter to `Dl=/usr/lib/lpdfilters/lps_40.a`.

Steps

The following figure depicts the system files that are created or changed when you are running the `lprsetup` command:



You must be logged in as superuser to run the lprsetup command.

Note

> To terminate lprsetup with no modifications to system files press CTRL/C.

1. Type the following:

   ```
   # lprsetup
   ```

2. Select the add option from the menu:

   ```
   ULTRIX Printer Setup Program

   Command  < add modify delete exit view quit help >: add
   ```

3. Optionally, enter additional names for the printer. To accept the default without specifying additional names, press the RETURN key.

   The lprsetup command uses an internal numbering scheme and, by default, assigns the next available number to the printer you are adding. It then automatically assigns the default number and lp*default-number* as printer names. The printer can be also be known by additional names.

4. Enter the type of printer you are adding, and then optionally indicate additional synonyms by which you want the printer known.

5. Accept the defaults or specify the appropriate values for the various printer parameters when lprsetup prompts you.

See Also

`tty`(4), `printcap`(5), `lpd`(8), `lprsetup`(8)

Guide to Ethernet Communications Servers
Guide to System Environment Setup

Establishing Disk Quotas

You can control the number of files and amount of disk space that each user can access by establishing disk quotas. The `edquota` command allows you to establish and modify disk quotas on individual users, and to use a prototypical user to establish the same quotas on a group of users. You can set quotas for both number of disk blocks and number of files.

Before You Start

Before running the `edquota` command, you should do the following:

- Verify that the QUOTA option is in your system configuration file

  The pathname of the system configuration file is `/usr/sys/conf/vax` or `/usr/sys/conf/mips`, depending on whether yours is a VAX or RISC machine. The system configuration filename is is the host name of the machine in uppercase.

  The QUOTA option is included in your system configuration by default. If it is not already there, include the following line in the options section of your system configuration file:

  ```
  options         QUOTA
  ```

  If you edit the system configuration file you must rebuild your kernel. Use the `doconfig` command and the `-c` option. The `-c` option specifies the use of the existing system configuration file to rebuild the kernel. The following command rebuilds the kernel of a system called `host1`:

  ```
  # doconfig -c HOST1
  ```

  See the `doconfig`(8) reference page for more information.

- Verify that the `quotaon -a` and `quotacheck -a` commands are in your `/etc/rc.local` file

  Both commands are included in the default `/etc/rc.local` file. Look for the following entries:

  ```
  echo -n ’check quotas: ’                 >/dev/console
          /etc/quotacheck -a
  echo ’done.’                             >/dev/console
  /etc/quotaon -a
  ```

- Edit the `/etc/fstab` file and change the mount type on the file systems for which you are establishing quotas to `rq`

  You can establish quotas only on file systems that are mounted locally and marked read-write.

  For example, the following `/etc/fstab` entry indicates that the `/usr` file system is read-write (`rw`):

  ```
  /dev/ra5h:/usr:rw:1:3:ufs::
  ```

  If you plan to establish quotas on the `/usr` file system, you must edit this entry to read as follows:

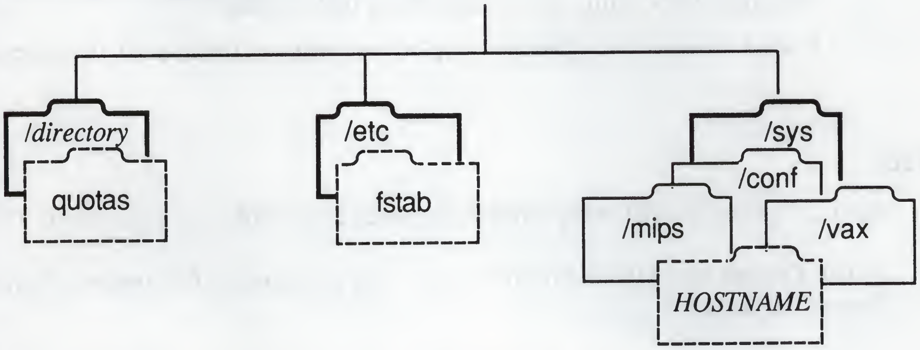
  ```
  /dev/ra5h:/usr:rq:1:3:ufs::
  ```

- Determine how much disk space you want to allot to each user

  The du command allows you to analyze disk usage. If you are establishing user partitions for the first time, you need to experiment with what amount of space is appropriate for each user. You can set quota on the number of blocks, the number of inodes, or both. See the du(1) reference page for more information.

Steps

The following figure depicts the system files that are created or changed when you establish disk quotas on your system:



You must be logged in as superuser to complete the following steps.

Complete step 1 if you are establishing quotas for the first time. If you are modifying quotas on a system where they are already established, start with step 2.

1. Bring your system to single-user mode by typing the following:

```
# shutdown now
```

2. Run the quotacheck command with the -f option.

   You must have a file named quotas at the top level of the file system in which you are establishing quotas. The -f option to the quotacheck command creates the quotas file.

   For example, if you are establishing quotas on the /usr file system, create a quotas file as follows:

```
# /etc/quotacheck -f /usr
```

3. Establish disk quotas with the edquota command.

   To establish or modify disk quotas for user1, do the following:

```
# edquota user1
```

   Edit the file that the edquota command creates with the quotas you want to apply to user1. Save the file.

   If you are modifying quotas that have already been established, just complete steps 2 and 3, as necessary.

4. Apply the same quotas that you established for user1 to a group of users.

   You can use user1 as a prototype for establishing quotas for a group of users as follows:

   ```
   # edquota -p user1 user2 user3 user4 user5
   ```

   The quotas that you established for user1 are now also established for user2, user3, user4, and user5. See the edquota(8) reference page for more information.

5. Reboot your system.

   Use the shutdown command with the -r option to reboot. The following command performs an orderly shutdown and automatic reboot, broadcasting the message "Rebooting after establishing disk quotas".

   ```
   # shutdown -r +5 "Rebooting after establishing disk quotas"
   ```

See Also

du(1), doconfig(8), edquota(8), quotacheck(8), quotaon(8), quotarep(8)

''Disk Quotas in a UNIX Environment,'' *Supplementary Documents, Volume 3: System Manager*

Network Setup 2

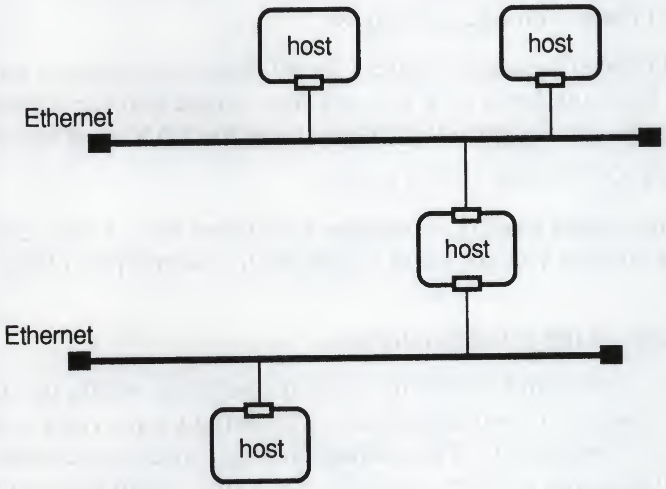
Local area networks (LANs) facilitate information exchange and resource sharing by linking together the machines at your site. Establishing a LAN requires careful planning based on many factors, including your current and projected networking needs, cost, reliability and maintainability, and geography.

The following tasks are described in this section:

- Setting up a network with `netsetup`
- Setting up the Network File System with `nfssetup`
- Setting up the Simple Network Management Protocol Agent with `snmpsetup`
- Setting up a router

The only mandatory task in this section is ''Setting Up a Network with `netsetup`.'' You must complete it before attempting the other tasks. Complete the other tasks in any order.

A variation on the following figure appears for each task. Each figure shows the hosts involved in completing the particular task and how they interact.



ZK-0169U-R

Setting Up a Network with netsetup

A local area network (LAN) is a group of two or more computer systems connected by a transmission medium. Each computer system, or host, is connected to the transmission medium by a hardware interface.

Every LAN should be assigned a unique network number by the Network Information Center (NIC). Every host connected to the LAN is assigned a number by the local network administrator that includes the LAN's network number and a host number unique to that host. All of the hosts on a particular LAN share the same network number.

The `netsetup` command automates establishing and adding nodes to a local area network (LAN). After your LAN is established, you can use the `netsetup` command to update the `/etc/hosts` and `/etc/hosts.equiv` files. See the `netsetup`(8) reference page for more information.

Before You Start

Before attempting to set up your network you should have obtained a network number from the NIC, and must have an understanding of TCP/IP networking concepts. For information on obtaining a network number, and a discussion of TCP/IP networking concepts, see the *Introduction to Networking and Distributed System Services*.

Before running `netsetup`, you should gather the following information:

- Your system's Internet address
- Your Internet Protocol broadcast address

  The Internet Protocol broadcast address for all hosts on a network must be the same. If you have any hosts on a network that require that the Internet Protocol broadcast address use all zeros, then all hosts on the LAN must use all zeros.
- Whether your LAN is using subnet routing

  If you are using subnet routing, determine how many bits of the `host` portion of the Internet address you are using to specify the subnet part of the network address.
- The device name of the network interface

  The `netsetup` command checks the system configuration file for the device name of your system's network interface. The default it provides is based on what devices are configured. If the default that `netsetup` provides is not correct, or if it does not provide a default, check the system configuration file or run the `netstat` command with the `-i` option to see what network interfaces are available to be configured.
- Your network name or alias

  Optionally, you can specify a name for your network. The `netstat` command uses the name you specify to translate the network address to the network name.
- The names and addresses of other hosts on the network

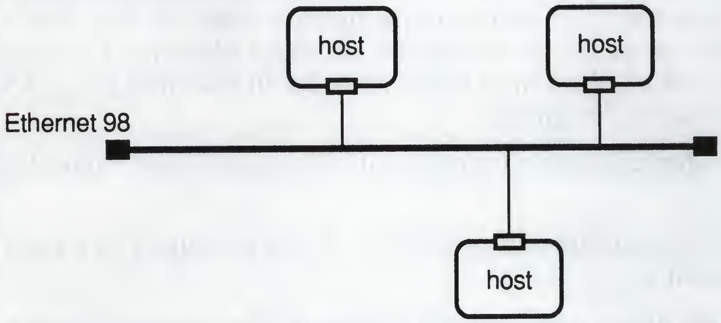
  Gather the names and Internet addresses of key hosts on your LAN. If the network is already established, you can copy the `/etc/hosts` file from a system that has a complete listing, rather than add each host on the network using `netsetup`.

- The names of trusted hosts

  Trusted hosts are listed in the /etc/hosts.equiv file. Systems listed in the /etc/hosts.equiv file are logically equivalent to, and therefore treated exactly the same as, the local system.

Steps

The following figure shows a LAN with three hosts attached. All of the hosts share the same network number, 98. Each host, however, is assigned a unique host number by the network manager.



ZK-0179U-R

You must be logged in as superuser to run the netsetup command.

Note

> To terminate netsetup with no modifications to system files press CTRL/C.

1. Type the following:

   ```
   # netsetup install
   ```

   The install option tells the system that this is a first-time installation.

2. Verify your system's name and optionally specify any abbreviations by which you want your system known.

   The netsetup command provides as the default the name that you specified for your system at installation. Following some informational text about abbreviations, it prompts you to specify one or more for your system.

   ```
   Your system's name is "host1".  Is this correct [yes]?
   ```

3. Specify the network number.

   Enter the network number that was assigned to you by the NIC. If you do not have a registered Internet number, specify the number that you are assigning to your LAN.

4. Indicate whether your network is using subnet routing.

   The subnet mask for your LAN is determined by how many bits from the host portion of the Internet address you use to specify subnet routing. All hosts on the same subnetwork must supply the same answer to this question.

5. Enter the host number.

   The host number must be unique to that host.

   The netsetup command then updates the appropriate system files with the information you have provided, displaying a message similar to the following:

   ```
   ***** UPDATING /etc/hosts WITH host1 AND localhost *****
   ```

6. If you answered yes in step 4 (you are using subnet routing), the netsetup command asks how many bits to use for specifying subnetworks. If you answered no in step 4, skip to to step 7.

   If your network uses subnet routing, all machines must specify the same number of bits from the host portion of the Internet address to use. The netsetup command determines the appropriate broadcast address and netmask for your system (both of which must be the same for all machines on a LAN) based on the information you provide.

7. Specify whether to use all zeros or all 1s for the Internet Protocol broadcast address.

   The industry standard default is all 1s. If you are setting up a LAN for the first time, use all 1s.

8. Specify the device name and unit number of your network interface.

   If the default that netsetup provides is incorrect, or if netsetup does not provide a default, check the system configuration file or run the netstat command with the -i option to see what network interfaces are available to be configured.

   The netsetup command then updates the appropriate system files with the information you have provided, displaying a message similar to the following:

   ```
   **UPDATING /etc/rc.local WITH network configuration information**
   ```

9. Specify a network name for your network number and any aliases for the network name.

   If you are adding your system to an existing network, specify the same name for the network as the other hosts on the network. If the network is new, you may want to name it based on function, or location, for example doconet for a network whose machines are involved in documentation.

   The netsetup command then updates the /etc/networks file with the name of the network, in this example doconet, displaying a message similar to the following:

   ```
   ** UPDATING /etc/networks WITH doconet **
   ```

10. Enter the host name, abbreviations, network number, and host number for each host on the network.

    The information you supply is used to update the /etc/hosts file. You should add the names of key hosts on your network to the /etc/hosts file, regardless of whether you intend to use BIND/Hesiod or Yellow Pages to distribute the hosts database on your network. See the *Guide to the BIND/Hesiod Service,* the *Guide to the Yellow Pages Service,* and the *Introduction to Networking and Distributed System Services* for information on distributing databases in a networked environment.

11. Enter the names of trusted hosts.

    Users on the trusted host who have a valid account on your machine can log in to your machine without supplying a password. Designate trusted hosts with care.

    The netsetup command then updates the appropriate system files with the information you have provided, displaying a message similar to the following:

    ```
    ***** SETTING UP /usr/hosts DIRECTORY *****

    ***** NETWORK SETUP COMPLETE *****
    ```

12. Reboot your system.

    Use the shutdown command with the -r option to reboot. The following command immediately performs an orderly shutdown and automatic reboot:

    ```
    # shutdown -r now
    ```

See Also

hosts(5), hosts.equiv(5), networks(5), ifconfig(8), netsetup(8), netstat(8)

Introduction to Networking and Distributed System Services

Setting Up the Network File System with nfssetup

The Network File System (NFS) is a facility for sharing files in a heterogeneous environment. It is based on the client/server model. An NFS server is a machine that exports file systems; an NFS client is a machine that imports file systems.

Your machine can be set up as an NFS server, an NFS client, or both.

Before You Start

Your network must be up and running before you attempt to set up NFS.

Before running the `nfssetup` command, you should gather the following information:

- Whether your system will be an NFS server, NFS client, or both
- Whether you want NFS locking enabled

  For file locking to work, it must be enabled on both clients and servers.

  File locking allows you to create advisory locks on local and remote files, and file regions. Locking prevents multiple users from editing the same file simultaneously. Advisory locking, however, is not enforced. For more information on file locking, see the *Guide to the Network File System* and the `fcntl`(2) and `lockf`(3) reference pages.
- The number of block I/O (`biod`) daemons you want to run

  The default number of 4 is adequate for an average workload. The maximum number of `biod` daemons configurable with `nfssetup` is 5.
- For servers, the number of `nfsd` daemons to run

  The default number of 4 is adequate for an average workload.

  The number of `nfsd` daemons should equal approximately 75 percent of the number of clients accessing the file system. The maximum number of `nfsd` daemons configurable with `nfssetup` is 20.
- Whether to run the `rwalld` daemon

  The `rwalld` daemon sends a broadcast message to clients when a server is shutting down using the `shutdown` command. If the server crashes, or is brought down with the `halt` or `reboot` command, `rwalld` does not send a broadcast message to clients.
- For servers, the directory pathnames of the directories that you want to export, and the host names of the machines to which you plan to export these directories

  If you want to limit what hosts can import a file system, you must specify the individual hosts or network groups explicitly in the `/etc/exports` file. If you do not specify individual hosts or network groups, all hosts can import that file system. For information on defining network groups, see the the `netgroup`(5yp) reference page and the *Guide to the Yellow Pages Service.*
- For clients, the remote host names (servers) from whom you are importing directories, the complete directory pathnames of the directories that you want to import, the local mount points where you want the imported directories to reside, and whether the imported file system should be read-only or read-write

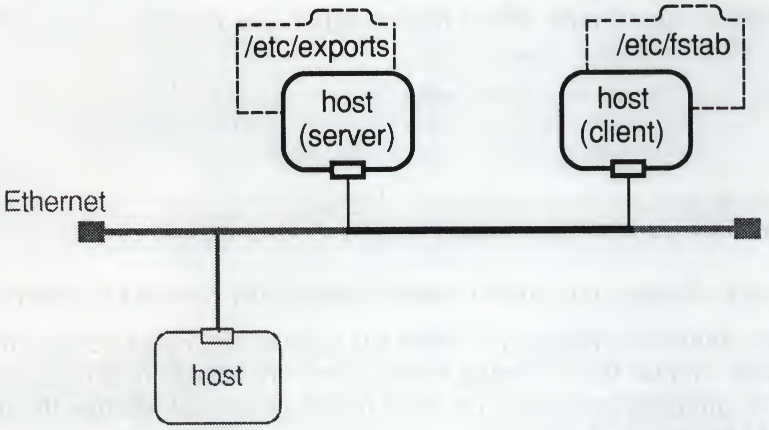
  The default permission is read-only.

Note

If you mount a file system on a client with read-write permissions (for example, a home directory), the user identification number (UID) for the owner of the file system on the server and the client must be the same. If they are not, the user cannot modify any of the mounted files.

Steps

The following figure shows an NFS server and an NFS client attached to a LAN. It indicates the system files that are modified when you run the nfssetup command. For hosts that are servers, the /etc/exports file is modified. For hosts that are clients, the /etc/fstab file is modified. The same host can be a server for some file systems and a client for others.



ZK-0180U-R

You must be logged in as superuser to run the nfssetup command.

Note

To terminate nfssetup with no modifications to any system files, press CTRL/C.

1. Type the following:

```
# nfssetup
```

2. Indicate whether you want NFS locking enabled.
3. Indicate whether you are exporting any directories.

   If you answer yes, the nfssetup command prompts you for the number of nfsd daemons to run.

   If you answer no, the nfssetup command skips to the next question.
4. Indicate the number of block I/O daemons to run.

5. Indicate whether you want to run the rwalld daemon.

   If you specified in step 3 that you are exporting directories, the nfssetup command next asks whether you want to add any directories to the /etc/exports file.

   If you specified in step 3 that you are not exporting directories, the nfssetup command next asks if you want to add any remote file systems to be mounted.

   If you are both exporting file systems and importing file systems, the nfssetup command asks you about modifying the /etc/exports file, and then about remote mounting file systems.

6. Indicate whether you want to add any directory pathnames to the /etc/exports file.

   If you choose to add any directory pathnames to the /etc/exports file the nfssetup command prompts you for the pathname of the directory to export, and what hosts or network groups to allow to import the file system.

   The following example shows how to export the directory /usr/users/jal to host1.cities.dec.com:

```
Enter the directory pathname: /usr/users/jal
        Netgroup/Machine name:  host1.cities.dec.com
        Netgroup/Machine name: RETURN

Enter the directory pathname: RETURN
Directory export list complete...
```

7. Indicate whether you want to mount (import) any remote file systems.

   If you choose to import any remote file systems the nfssetup command prompts you for the following information: the remote host name (server), the remote directory pathname, the local mount point, and whether the file system should be imported read-only.

   The following example shows how to mount the directory /usr/projects from host3 onto the local mountpoint /usr/staff/projects, and to assign read-write permissions to it.

```
Enter the remote host name: host3

        Enter the remote directory pathname: /usr/projects
        Enter the local mount point: /usr/staff/projects
        Is this a read-only mount [y] ? n

        Enter the remote directory pathname: RETURN

Enter the remote host name: RETURN
Remote directory mount list complete...
```

   If you specify a local mount point that does not exist, the nfssetup command creates it.

8. Confirm (**c**) the information you have entered, quit (**q**) nfssetup with no changes, or restart (**r**) the procedure.

   If you choose **c**, the nfssetup command displays information similar to the following:

```
Updating files:
        /etc/rc.local
        /etc/fstab
        /etc/exports
```

9. Press RETURN when the nfssetup command asks if you want to start the NFS daemons automatically:

```
Would you like nfssetup to start the daemons
                                 automatically [y]? RETURN
```

   If you do not have nfssetup start the daemons automatically, you must start them manually. See the *Guide to the Network File System* for information on starting the NFS daemons manually.

   The nfssetup command then displays the following instructional text:

```
In order to mount the remote directories you wish to access,
type the following command after exiting from nfssetup:

        # /etc/mount -a -t nfs
```

   Running the /etc/mount command prevents you from having to reboot your machine to access imported directories.

See Also

fcntl(2), lockf(3), nfssetup(8nfs)

Guide to the Network File System
Guide to the Yellow Pages Service

Setting Up the Simple Network Management Protocol Agent with snmpsetup

The Simple Network Management Protocol (SNMP) is the de facto industry standard for managing TCP/IP networks. The protocol defines the role of a Network Management Station (NMS) and an SNMP Agent, allowing remote users on an NMS to monitor and manage TCP/IP network entities.

The `/etc/snmpd.conf` file is the configuration file for the SNMP daemon, `snmpd`.

Note

The ULTRIX software supports an implementation of the SNMP Agent. It does not implement the NMS software.

Before You Start

Your network must be up and running before you attempt to set up SNMP.

Before running the `snmpsetup` command, you should gather the following information:

- The names of your system's network interfaces

  Check the system configuration file for the device name of your system's network interfaces, or run the `netstat` command with the `-i` option to see what network interfaces are available to be configured. See the `netstat`(8) reference page for more information.

- Your community name and type

  The community name can be any character string up to 127 characters long.

  Communities can be read-only, read-write, or traps. Read-only communities can be monitored but not managed by an NMS. Read-write communities can be managed by the NMS. Traps are unsolicited messages generated by the Agent that guide the polling from the NMS.

- The Internet address that you want associated with the community

  You will need to specify the Internet address of any NMS that you want to be able to manage or monitor your system.

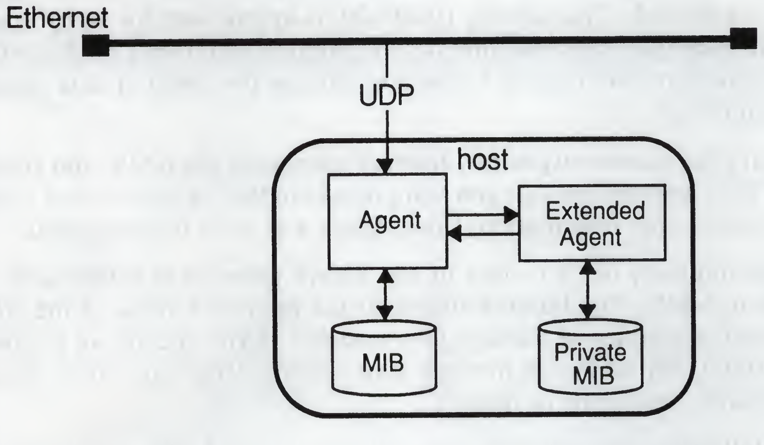
- Whether you are including user-written Extended Agents

  If you are defining Extended Agents, you must specify the full pathname of the extended agent and its name.

  See the *Guide to Network Programming* for information on defining Extended Agents, and the directory `/usr/examples/snmp/snmpext` for an example.

Steps

The following figure shows a host running the SNMP Agent software. It shows the relationship between the Agent and the Management Information Base (MIB), and the relationship between the Agent and the Extended Agent. The NMS (not shown) can be located either on the same network, or a different one.



ZK–0181U–R

You must be logged in as superuser to run the `snmpsetup` command.

Note

To terminate `snmpsetup` with no modifications to the `/etc/snmpd.conf` file, press CTRL/C.

1. Type the following:

```
# snmpsetup
```

2. Press the RETURN key to accept the default `sysDescr` parameter.

   The default `sysDescr` parameter is the system name that you specified at installation. Digital recommends that you change the default `sysDescr` parameter only if you want to add system hardware and software information.

3. Add network interfaces that are not automatically configured by the SNMP Agent.

   The SNMP Agent checks the system configuration file and automatically configures the following interfaces, if they are present: de, ln, lo, ni, qe, scs, and xna. No explicit entries for these interfaces appear in the `/etc/snmpd.conf` file.

   If you are configuring a network interface other than one of the ones listed, `snmpsetup` prompts you for information about the interface, and then edits the `/etc/snmpd.conf` file with the appropriate information.

   The following example shows how to add a serial-line (sl0) interface, and how to specify the interface type (`ifType`) and interface speed (`ifSpeed`).

```
Do you wish to add network interfaces [n]? y
Enter new interface name (ifName)? sl0
Enter interface type (ifType) [6]? 1
Enter interface speed (ifSpeed) [10000000]? 9600
```

The ifType parameter indicates the code number for the proper interface hardware type. The value of 1, specified in this example, indicates that serial-line interfaces belong in the category other. See the snmpd.conf(5) reference page or RFC 1066 under the ifType object definition for more information on coding interface hardware types.

The ifSpeed parameter is an estimate of the interface's current bandwidth in bits per second. The default, 10000000, is appropriate for an Ethernet interface. In this example, the serial line is configured to run over a modem at 9600 baud. See your hardware manual for information on the speed of data transmission for your interface.

4. Specify the community name, Internet address of the NMS, and community type for communities that you want to add to the /etc/snmpd.conf file. The community information is mandatory and must be configured.

   The community name is used by the SNMP protocol to authenticate requests from an NMS. The Internet address is the Internet address of the NMS that is allowed to monitor or manage your system. If you specify an Internet address of 0.0.0.0, any NMS can monitor your system. The community type can be read-write, read-only, or traps.

   The following example shows how to set up a read-only and a read-write community:

   ```
   Enter community name? test1
   Enter IP address associated with community
                           test1 [0.0.0.0]? 128.45.10.100
   Select community type (read-only,read-write,
                           traps) [read-only]? RETURN

   Do you wish to add another community [n]? y
   Enter community name? testwrite
   Enter IP address associated with community
                           testwrite [0.0.0.0]? 128.45.12.105
   Select community type (read-only,read-write,
                           traps) [read-only]? read-write
   ```

   The community test1 is a read-only community that allow the NMS whose Internet address is 128.45.10.100 to monitor it. The community testwrite is a read-write community that allows the NMS whose Internet address is 128.45.12.105 to both monitor it and set variables for it.

5. Press the RETURN key when prompted to configure a public read-only community, if you have not configured any other communities.

   You must configure a public read-only community if you have not defined any other communities. If you do not have any community names configured, SNMP will not work on your system.

6. Specify the full pathname and name of any user-written Extended Agents, if you are adding any.

   The *Guide to Network Programming* has information on defining Extended Agents. You should follow the steps described there before specifying Extended Agents using snmpsetup.

After you have finished answering the questions about Extended Agents, snmpsetup exits.

See Also

`snmpd.conf`(5), `netstat`(8), `snmpd`(8), `snmpsetup`(8)

Introduction to Networking and Distributed System Services
Guide to Network Programming
RFC 1065—*Structure and Identification of Management Information for TCP/IP-based internets*
RFC 1066—*Management Information Base for Network Management of TCP/IP-based internets*
RFC 1098—*A Simple Network Management Protocol (SNMP)*

Setting Up a Router

An Internet network consists of two or more local area networks connected by a computer system that acts as a router (commonly called a gateway).

Routers are hosts that are connected to multiple LANs. They have a network interface for each LAN to which they are connected, and each network interface is assigned a unique host name and Internet address. Because it is connected to multiple LANs, a router allows data to be transferred between systems on the LANs to which it is connected.

Before You Start

Any networks that you intend to interconnect with the router must be up and running.

Before setting up your system as a router, you should gather the following information:

- An understanding of Internet addresses, netmasks, and subnetworks

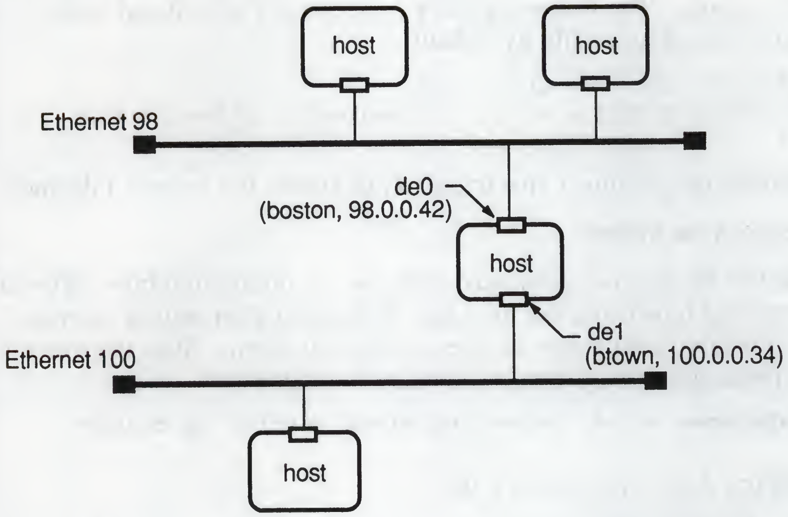
  For a discussion of TCP/IP LANs, Internet addresses, and subnetworks, see the *Introduction to Networking and Distributed System Services*.

- Whether the appropriate network interfaces are installed

  Your system must have a network interface for each network to which it is connected. Check the system configuration file or run the `netstat` command with the `-i` option to see what interfaces are available to be configured.

Steps

The following figure shows two LANs interconnected by a router. The router host has network interfaces for each of the networks it is connected to. On network 98 the router is known as `boston` and has an Internet address of 98.0.0.42. On network 100 the router is known as `btown` and has an Internet address of 100.0.0.34.



ZK–0178U–R

You must be logged in as superuser to complete the following steps:

1. Assign a unique name (pseudo-hostname) and address to the new network interface and edit the `/etc/hosts` file with the pseudo-hostname and address.

   Although a router is one physical machine, it functions as multiple hosts. Each network interface is assigned a unique hostname and Internet address in accordance with the numbering and address scheme of the network to which it is directly connected.

   For example, a system that is a router between networks 98 and 100 can be known on network 98 as `boston` and have an Internet address of 98.0.0.42. Its pseudo-hostname on network 100 can be `btown` with an Internet address of 100.0.0.34.

2. Edit the `/etc/rc.local` file with a line that has the following format:

   `/etc/ifconfig` *device-name pseudo-hostname* `broadcast` *x.x.x.x* `netmask` *y.y.y.y*

   The *device-name* is the network interface. The *pseudo-hostname* is the host name associated with the new network interface. The `broadcast` indicates the Internet broadcast address. The `netmask` tells the system whether subnetworks are in use.

   In the figure `btown` is connected to network 100 by a DEUNA (de1) controller. Place the entry for the new network interface after the line for the primary network interface, and before the entry for `localhost`. The corrected edited `/etc/rc.local` file should look similar to the following:

```
/etc/ifconfig de1 btown broadcast 100.255.255.255 netmask 255.0.0.0
```

3. Enable the routed daemon.

   When the routed daemon is running, the internal routing tables are updated periodically. The following entry for routed is included in the /etc/rc.local file by default:

   ```
   #if [ -f /etc/routed ]; then
   #    /etc/routed & echo -n ' routed'    >/dev/console
   #fi
   ```

   Remove the comment characters (#) to enable the routed daemon.

4. Reboot your system.

   Use the shutdown command with the -r option to reboot. The following command broadcasts the message "Rebooting after setting up router" to all users at intervals starting five minutes before shutdown. After the five minutes, it performs an orderly shutdown and automatic reboot:

   ```
   # shutdown -r +5 "Rebooting after setting up router"
   ```

5. Edit the /etc/networks file.

   The /etc/networks file allows the netstat command to translate a network number into a network name. An /etc/networks file that has been edited to include the new network (network 100) contains the following entries:

   ```
   #
   # Internet networks
   #
   loop          127    loopback
   ethernet1      98    doconet
   ethernet2     100    devonet
   ```

   The first field is the network name. For example, network 100 is also known as ethernet2. The second field is the network number, and the third field specifies any network alias names. See the netstat(8) reference page for more information.

Note

> If you choose to run either Yellow Pages (YP) or the BIND/Hesiod naming service, the networks database is distributed. You need only edit the networks database on the YP or BIND/Hesiod master server, and the information is distributed to all other servers and client systems. For more information on distributing databases with BIND/Hesiod, see the *Guide to the BIND/Hesiod Service*. For more information about distributing databases with YP, see the *Guide to the Yellow Pages Service*.

To access the new router, system administrators of each host on the networks to which the router is connected must do the following:

1. Edit the /etc/networks file.

   See step 5 above.

2. Enable the routed daemon and optionally reboot the system.

   See steps 3 and 4 above.

To update the tables immediately without rebooting, start `routed` manually. As superuser, type the following:

```
# /etc/routed
```

If you choose not to run the `routed` daemon at all, you can add a new route using the `/etc/route` command. The syntax for the `/etc/route` command is as follows:

```
/etc/route command [ net | host ] destination_router [ metric ]
```

A route added using the `/etc/route` command is effective until the system is rebooted. See the *Introduction to Networking and Distributed System Services* and the `route`(8c) reference page for more information on adding routes manually.

See Also

`netstat`(1), `ifconfig`(8c), `route`(8c), `routed`(8c)

Introduction to Networking and Distributed System Services
Guide to the BIND/Hesiod Service
Guide to the Yellow Pages Service

Distributed System Services Setup 3

The ULTRIX software supports the following distributed system services:

- BIND/Hesiod and Yellow Pages naming services (which can be used either singly or in combination)
- Network Time Protocol (NTP) and Time Synchronization Protocol (TSP) time services
- Kerberos authentication service (see the *Guide to Kerberos* for information on Kerberos)

Additionally, there are three configurable security modes: BSD (the default), UPGRADE and ENHANCED.

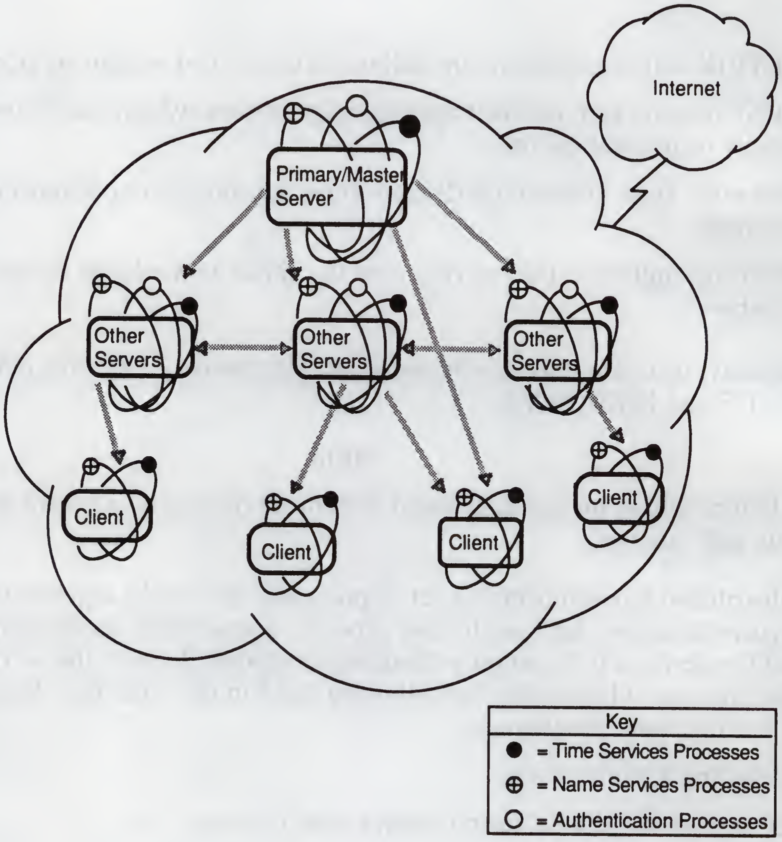
Note

> Before setting up any distributed system services, your network must be up and running.

Your distributed environment is a set of processes that works together to coordinate time synchronization, database lookup services, and network security on your LAN. Each of the services is based on a client/server model. Because the services work together, you should complete the following tasks in the order that they are described if this is a first-time installation:

- Selecting a name service
- Setting up the BIND/Hesiod service with `bindsetup`
- Setting up the Yellow Pages service with `ypsetup`
- Setting up the `svc.conf` file with `svcsetup`
- Setting up the network time services
- Adding users in a distributed environment

The following figure depicts a distributed environment that is running processes for naming services, time services, and authentication services. It illustrates, in a general way, the relationship between the master or primary servers, other servers, and clients in a distributed environment. If possible, designate the same machine as the master or primary server for all of the services. The gray arrows indicate the flow of data between machines.



ZK–0173U–R

Selecting a Name Service

The ULTRIX software supports the BIND/Hesiod and Yellow Pages (YP) naming services. Depending on your network and your security needs, you can run one or both of them. You can also choose not to run a naming service at all.

Both BIND/Hesiod and YP are based on a client/server model, and both enable you to coordinate the distribution of information on your LAN. However, they do not provide exactly the same functionality.

Using BIND/Hesiod – BIND/Hesiod distributes the following databases:

`aliases`
`auth`
`group`
`hosts`
`networks`
`passwd`
`protocols`
`rpc`
`services`

The Berkeley Internet Name Domain (BIND) service organizes the entire Internet hierarchically, and provides domain name-to-Internet address mapping (or resolution) for hosts throughout the Internet. At the top of the BIND hierarchy are seven root name servers that recognize the top level domains (for example, `com`, `gov`, `mil`, and `org`). The top-level domains are further divided into subdomains. When you register your network with the Network Information Center (NIC), it assigns your network a unique number and domain name. If you decide to use BIND to distribute the `hosts` database and are connected to the Internet, you must use the BIND domain name assigned to your network by the NIC.

Your LAN must be connected to the Internet to take advantage of the Internet-wide host name-to-address resolution functionality of BIND, although you can also use BIND to resolve host names and addresses within your LAN.

Hesiod is layered on top of BIND, and enables you (within your LAN) to distribute all the other databases besides `hosts`. You can also write your own Hesiod application and serve your own Hesiod database. For information on writing and distributing a Hesiod application, see the *Guide to the BIND/Hesiod Service.*

If you are concerned about server spoofing (where a machine that is not a server masquerades as one that is, and distributes false information), you can run the Kerberos-authenticated `named` daemon on your BIND/Hesiod servers. Kerberos, an authentication service, guarantees the authenticity of the data that the BIND/Hesiod servers return.

Using Yellow Pages – YP distributes the following databases:

```
aliases     passwd
group       protocols
hosts       rpc
netgroup    services
networks
```

It also allows you to serve specific fields of the `passwd` database LAN-wide, while designating other fields locally. For example, the following entry in the `passwd` database indicates that all fields but the login shell field should be derived from the master `passwd` database. The login shell for this user on the local machine should be `/usr/new/csh`.

```
+gene::::::/usr/new/csh
```

YP also organizes the hosts on your LAN into a domain. However, the meaning of a YP domain differs from that of BIND/Hesiod. With YP, your domain is a local, site-specific, administrative entity whose name is chosen by the local network administrator. YP resolves queries only within your LAN.

Using Neither Naming Service – If you choose not to run a naming service at all, you must maintain all of the databases individually on each machine as local `/etc` files.

Before You Start

Before selecting which name services to run on your LAN, you should gather the following information:

- Determine your security needs

  Although security issues are beyond the scope of this manual, they are important when you are deciding which name service to run.

  If you want to take advantage of the enhanced security features supported by the ULTRIX software, your environment must be a homogeneous environment (all hosts running ULTRIX Version 4.0), and you must use BIND/Hesiod to distribute the `passwd` and `auth` databases. Also, you must run BIND/Hesiod servers with Kerberos to prevent server spoofing, and to authenticate the enhanced security features. See the *Guide to Kerberos* for more information on the Kerberos authentication service.

- Whether your network is heterogeneous or homogenous

  Your network configuration places constraints on what databases you can serve to which hosts. A heterogeneous environment is one in which other vendors' machines run YP or BIND, or machines run ULTRIX prior to version 4.0. All implementations of YP, regardless of vendor, are compatible, as are all implementations of BIND. However, if your environment has hosts running ULTRIX Version 4.0, those hosts can only use Hesiod to serve databases within your LAN to machines that are also running Hesiod.

- Determine which databases you want to distribute, and with which name service

  For the most part, BIND/Hesiod and YP distribute the same databases. You can distribute some databases with BIND/Hesiod and some with YP if you have both services running on your LAN.

  However, if your LAN is connected to the Internet and you want to be able to resolve host names and addresses across the Internet, you must use BIND/Hesiod to distribute the `hosts` database. If your LAN is not connected to the Internet, both BIND/Hesiod and YP provide good host name and address resolution within your LAN.

  Also, you must use BIND/Hesiod to distribute the `auth` database if you are using the ULTRIX software's enhanced security features.

  Only YP distributes the `netgroup` database, which defines network-wide groups used for permission checking when doing remote mounts, remote logins, and remote shells.

The following table summarizes the features of each naming service, and the circumstances under which you would run each one:

Functionality	**Name Service**		
	YP	**BIND**	**BIND/Hesiod**
Enhanced security			Yes
Serves databases in a homogeneous environment	Yes	Yes	Yes
Serves databases in a heterogeneous environment	Yes	Yes	
Wide area network connectivity		Yes	Yes
Serves `netgroup` database	Yes		
Serves specific `passwd` fields in `passwd` database	Yes		

Steps

For information on setting up BIND/Hesiod on your LAN, see the task ''Setting Up BIND/Hesiod with bindsetup.'' For information on setting up YP on your LAN, see the task ''Setting Up the Yellow Pages Service with ypsetup.''

See Also

Guide to the BIND/Hesiod Service
Guide to Kerberos
Guide to the Yellow Pages Service

Setting Up the BIND/Hesiod Service with bindsetup

The BIND/Hesiod Service is a naming service that allows you to distribute the following network-wide databases: `aliases`, `auth`, `groups`, `hosts`, `networks`, `passwd`, `protocols`, `rpc`, and `services`.

The BIND/Hesiod service is based on a client/server model. Databases are maintained on the primary server, and updated information is distributed to secondary and slave servers. Caching servers have access to the Internet, but do not maintain databases. Instead, they service queries by asking other servers for the information, and then storing the answers they receive. Clients query a server for information.

The `bindsetup` command automates setting up the BIND/Hesiod service on your system.

Before You Start

Before running the `bindsetup` command, you should gather the following information:

- The role each host will play in your distributed environment

  You must choose one host to be the primary server, one or more hosts to be secondary and slave servers, and (optionally) a host to be a caching server. The rest of the hosts should run as BIND/Hesiod clients. For more information on the organization of the BIND/Hesiod service and the role each of the servers plays, see the *Guide to the BIND/Hesiod Service*.

- Your default domain name

  The Network Information Center (NIC) assigns a default domain name when you register for an Internet network number. If you are not connected to the Internet and never plan to be, you can choose your own default domain name.

- For the primary server, the databases you want to distribute

  If you want `bindsetup` to create the BIND/Hesiod database files for the databases you plan to distribute, you must copy the `/etc`-style source files for each database to the `/var/dss/namedb/src` directory.

- For secondary and slave servers, the host name and Internet address of the primary server, and one or more secondary servers

- Whether you want to set up any caching servers

  Caching servers have Internet access but do not maintain databases. For security reasons, you may want to use caching servers as routers (commonly called gateways) to the Internet rather than using primary or secondary servers as routers.

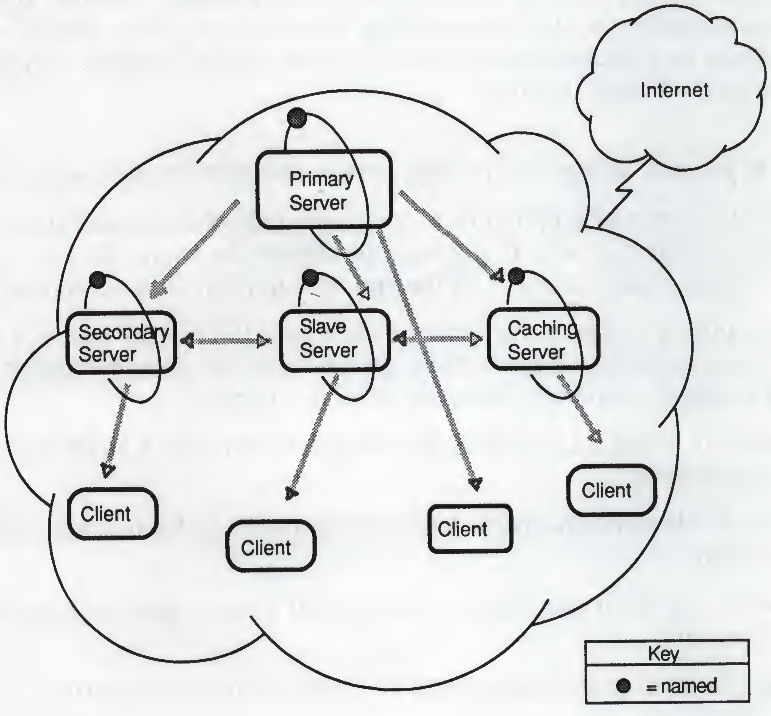
- For clients, the host name and Internet address of at least one server

- Whether you intend to run the Kerberos authentication service

  You can use the bindsetup command to configure a Kerberos-authenticated BIND/Hesiod server on your system. For information on setting up Kerberos, see the *Guide to Kerberos*.

Steps

The following figure depicts a distributed environment that is running BIND/Hesiod. It illustrates the processes running on each host and the relationship between the primary server, other servers, and clients. The gray arrows indicate the flow of data between hosts.



ZK-0172U-R

You must be logged in as superuser to run the bindsetup command.

Note

To terminate the bindsetup command with no modifications, press CTRL/C.

1. Type the following:

```
# bindsetup
```

2. Select the add option from the configuration menu:

```
Berkeley Internet Name Domain (BIND)
        Action Menu for Configuration

        Add                => a
        Modify             => m
        Remove             => r
        Exit               => e

Enter your choice [a]: RETURN
```

3. Enter the default domain name supplied by the NIC.

If you are setting up a primary server go to "Primary Server" section. If you are setting up a secondary or slave server go to "Secondary or Slave Server" section. If you are setting up a caching server go to "Caching Server" section. If you are setting up a client go to "Client" section.

Primary Server – If you are setting up a primary server, complete the following steps:

1. Select the `primary` option from the configuration menu, and answer `yes` when `bindsetup` asks if you want to convert the source files in `/var/dss/namedb/src` to the appropriate BIND/Hesiod format.

   If you answer `no`, or if the `/var/dss/namedb/src` directory is empty, `bindsetup` edits the appropriate system files, but prints a warning that you must create the database files once setup is complete.

2. Answer `no` when `bindsetup` asks if you want to run a Kerberos-authenticated `named` daemon.

   If you decide to run Kerberos later, see the *Guide to Kerberos* for setup information.

3. Answer `yes` when `bindsetup` asks if you want to start the `named` daemon automatically.

   The `bindsetup` command starts the `named` daemon and exits.

After `bindsetup` is finished, you must edit the `/etc/svc.conf` file. See the task ''Setting Up the svc.conf File with svcsetup'' for information on editing the `svc.conf` file.

Secondary or Slave Server – If you are setting up a secondary or slave server, complete the following steps:

1. Select the `secondary` or `slave` option from the configuration menu.

   The setup for secondary and slave servers is the same, although the servers function differently.

2. Enter the host name and Internet address of the primary server for your domain.

3. Answer `no` when `bindsetup` asks if you want to run a Kerberos-authenticated `named` daemon.

   If you decide to run Kerberos later, see the *Guide to Kerberos* for setup information.

4. Answer yes when bindsetup asks if you want to start the named daemon automatically.

   The bindsetup command starts the named daemon and exits.

After bindsetup is finished, you must edit the /etc/svc.conf file. See the task ''Setting Up the svc.conf File with svcsetup'' for information about editing the svc.conf file.

Caching Server – If you are setting up a caching server, complete the following steps:

1. Select the caching option from the configuration menu.
2. Answer no when bindsetup asks if you want to run a Kerberos-authenticated named daemon.

   If you decide to run Kerberos later, see the *Guide to Kerberos* for setup information.
3. Answer yes when bindsetup asks if you want to start the named daemon automatically.

   The bindsetup command starts the named daemon, and exits.

After bindsetup is finished, you must edit the /etc/svc.conf file. See the task ''Setting Up the svc.conf File with svcsetup'' for information about editing the svc.conf file.

Client – If you are setting up a client, complete the following steps:

1. Select the client option from the configuration menu.

   There must be a primary server already configured for the domain before any clients run bindsetup.
2. Enter the host name and Internet address of at least one server for your domain.

   It is recommended that you enter the primary server and one or more secondary servers.

After bindsetup is finished, you must edit the /etc/svc.conf file. See the task ''Setting Up the svc.conf File with svcsetup'' for information on how to edit the svc.conf file.

See Also

bindsetup(8), svcsetup(8)

Introduction to Networking and Distributed System Services
Guide to the BIND/Hesiod Service

Setting Up the Yellow Pages Service with ypsetup

The Yellow Pages (YP) service provides a distributed data lookup service for sharing information between systems on a local area network (LAN). YP allows you to distribute the following network-wide databases: `aliases`, `group`, `hosts`, `netgroup`, `networks`, `passwd`, `protocols`, `rpc`, and `services`.

YP is based on a client/server model. Database files, or **maps**, are located in `/var/yp/`*domainname*, and are stored and maintained on a master server. Changes to the database files are propagated at regular intervals to the slave servers. Clients do not store databases locally; they query servers for information.

The `ypsetup` command automates setting up YP on your system.

Before You Start

Before running the `ypsetup` command, you should gather the following information:

- Your default domain name

  A YP domain is an administrative entity that is organized into a master server, one or more slave servers, and numerous clients. The domain name that you choose can be any string of alphanumeric characters that is 31 characters or less in length. All systems in the domain must declare the same domain name.

- The role each host will play in your distributed environment

  Select one host to be the master server. There can be only one master server for each domain. Select one or more hosts to be slave servers. The rest of the hosts should run as YP clients.

  For more information on the organization of YP, and the roles each of the servers plays, see the *Guide to the Yellow Pages Service*.

- Whether you want to lock the `ypbind` daemon to a particular domain name and server list

  Normally, hosts broadcast YP requests on the network and the first available server answers the request. The `-S` option allows the you to lock the `ypbind` daemon to a particular domain and set of servers. Requests are made directly to the specified servers, rather than being broadcast. Digital recommends that you run YP with the `-S` option configured.

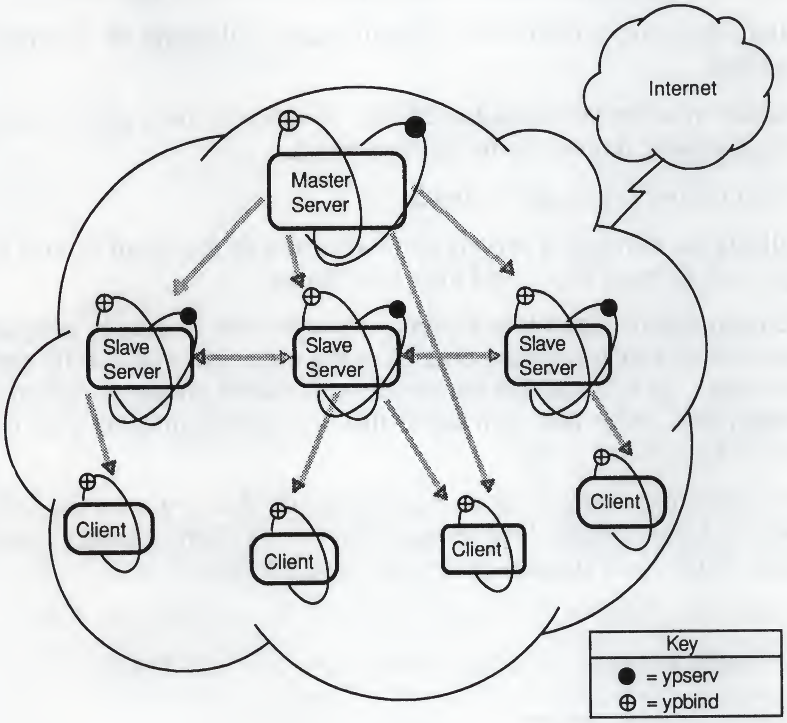
  If you choose to run YP with the `-S` option configured, you must know the host names of the servers to which you are locking the `ypbind` daemon.

- Whether to run the `yppasswdd` daemon

  The `yppasswdd` daemon allows the master copy of the password file to be updated remotely.

Steps

The following figure depicts a distributed environment that is running YP. It illustrates the processes running on each host and the relationship between the master server, slave servers and clients. The gray arrows indicate the flow of data between hosts.



ZK–0182U–R

You must be logged in as superuser to run the ypsetup command.

Note

> To terminate ypsetup with no modifications to system files press CTRL/C.

1. Type the following:

   ```
   # ypsetup
   ```

2. Enter the default domain name.

   The domain name can be any combination of letters and numbers. All systems in the domain must enter the same domain name.

3. Select whether you are configuring a master server (**m**), slave server (**s**), or a client (**c**).

If you are establishing a YP domain for the first time, you must configure the master server first.

Master Server – If you are setting up a master server, complete the following steps:

1. Enter `yes` or `no` when asked if you want to run the `yppasswdd` daemon.
2. List the names of other systems that will be configured as servers.

   The hosts that you specify must be listed in the `/etc/hosts` file. These hosts automatically receive updated versions of the `hosts` database.

   After `ypsetup` initializes the domain maps, it displays an informational message.
3. Indicate whether you want to add the `-S` option to the `ypbind` daemon to lock it to a specific domain name and server list.

   If you choose not to, go to step 5.
4. Indicate the number of servers that will make up the set of servers to which the `ypbind` daemon locks, and their host names.

   You can specify up to four servers, although three is usually adequate. The servers that you specify are queried in the order that you specify them. Therefore, on systems that are servers you should always specify the local system first. Note that each server that you specify must have an entry in the local `/etc/hosts` file.

   The following example shows how to specify that you want the `ypbind` daemon locked to the three servers `server1` (where `server1` is the host name of the local system), `server2`, and `server3`:

   ```
   Would you like to add the -S option to ypbind [n] ? y

   How many servers do you wish to specify [1] ? 3

   Server 1 name: server1

   Server 2 name: server2

   Server 3 name: server3
   ```
5. Answer `yes` when `ypsetup` asks if you want to start the YP daemons automatically.

After `ypsetup` starts the daemons, it displays an informational message reminding you to edit the `/etc/svc.conf` file with the order in which you want the name services queried for each distributed database. Then it exits.

See the task ‘‘Setting Up the svc.conf File with svcsetup’’ for information on editing the `svc.conf` file.

Slave Server – If you are setting up a slave server, complete the following steps:

Note

> Before configuring your system as a slave server, be sure that there is a master server configured for your domain, that you know its name, and that it is up and running.

1. Specify the name of the master server for your domain.

   The `ypsetup` command displays a message that it is copying the YP maps from the master server.

   After `ypsetup` transfers the domain maps, it displays an informational message.

2. Indicate whether you want to add the `-S` option to the `ypbind` daemon to lock it to a specific domain name and server list.

   If you choose not to, go to step 4.

3. Indicate the number of servers that will make up the set of servers to which the `ypbind` daemon locks, and their host names.

   You can specify up to four servers, although three is usually adequate. The servers that you specify are queried in the order that you specify them. Therefore, on systems that are servers you should always specify the local system first. Note that each server that you specify must have an entry in the local `/etc/hosts` file.

   The following example shows how to specify that you want the `ypbind` daemon locked to the three servers `server1` (where `server1` is the name of the local system), `server2`, and `server3`:

   ```
   Would you like to add the -S option to ypbind [n] ? y

   How many servers do you wish to specify [1] ? 3

   Server 1 name: server1

   Server 2 name: server2

   Server 3 name: server3
   ```

4. Answer `yes` when `ypsetup` asks if you want to start the YP daemons automatically.

After `ypsetup` starts the daemons, it displays an informational message reminding you to edit the `/etc/svc.conf` file with the order in which you want the name services queried for each distributed database. Then it exits.

See the task ‘‘Setting Up the svc.conf File with svcsetup’’ for information on editing the `svc.conf` file.

Client – If you are setting up a client, complete the following steps:

Note

> Before configuring your system as a client, be sure that there is at least one master or slave server configured for your domain.

1. Enter a **c** when the `ypsetup` command prompts you to be sure that a server is configured for your domain.
2. Indicate whether you want to add the `-S` option to the `ypbind` daemon to lock it to a specific domain name and server list.

   If you choose not to, go to step 4.
3. Indicate the number of servers that will make up the set of servers to which the `ypbind` daemon locks, and their host names.

   You can specify up to four servers, although three is usually adequate. The servers that you specify are queried in the order that you specify them. Note that each server that you specify must have an entry in the local `/etc/hosts` file.

   The following example shows how to specify that you want the `ypbind` daemon locked to the three servers `server1`, `server2`, and `server3`:

   ```
   Would you like to add the -S option to ypbind [n] ? y

   How many servers do you wish to specify [1] ? 3

   Server 1 name: server1

   Server 2 name: server2

   Server 3 name: server3
   ```
4. Answer `yes` when `ypsetup` asks if you want to start the YP daemons automatically.

After `ypsetup` starts the daemons, it displays an informational message reminding you to edit the `/etc/svc.conf` file with the order in which you want the name services queried for each distributed database. Then it exits.

See the task ‘‘Setting Up the svc.conf File with svcsetup’’ for information on editing the `svc.conf` file.

For All Machines Running YP – If YP is serving either the `/etc/passwd` or `/etc/group` file (or both), you must add the character sequence `+:` to the end of the appropriate database file after `ypsetup` exits.

The following example shows an `/etc/passwd` file that has been edited:

```
#  /etc/passwd file that is served by YP
#
#
#
root:M11slKjPj59vA:0:1:System PRIVILEGED Account:/:/bin/csh
field:e1uAN/FcWqZg.:0:1:Fld Svc PRIVILEGED Account:/usr/field:/bin/csh
nobody:Nologin:-2:-2:anonymous NFS user:/:
operator:PASSWORD HERE:0:28:Operator PRIVILEGED Account:/opr:/opr/opser
ris:Nologin:11:11:RIS Account:/usr/adm/ris:/bin/sh
daemon:*:1:1:Mr Background:/:
sys:PASSWORD HERE:2:3:Mr Kernel:/usr/sys:
+:
```

See Also

`domainname`(1yp), `ypfiles`(5yp), `svcsetup`(8), `ypbind`(8yp), `yppasswd`(8yp), `ypserv`(8yp), `ypsetup`(8yp), `ypwhich`(8yp)

Guide to the Yellow Pages Service

Setting Up the svc.conf File with svcsetup

The /etc/svc.conf file is the database service selection and security configuration file. It enables you to specify for each database the order in which the name services running on your system should be queried.

The svcsetup command automates modifying the /etc/svc.conf file. You need to run the svcsetup command after setting up the naming services on your system.

Before You Start

Before running the svcsetup command, you should gather the following information:

- Which distributed system services you are planning to run

  See the *Introduction to Networking and Distributed System Services* for information on planning a distributed environment.

- Whether your environment is heterogeneous (multivendor) or homogeneous

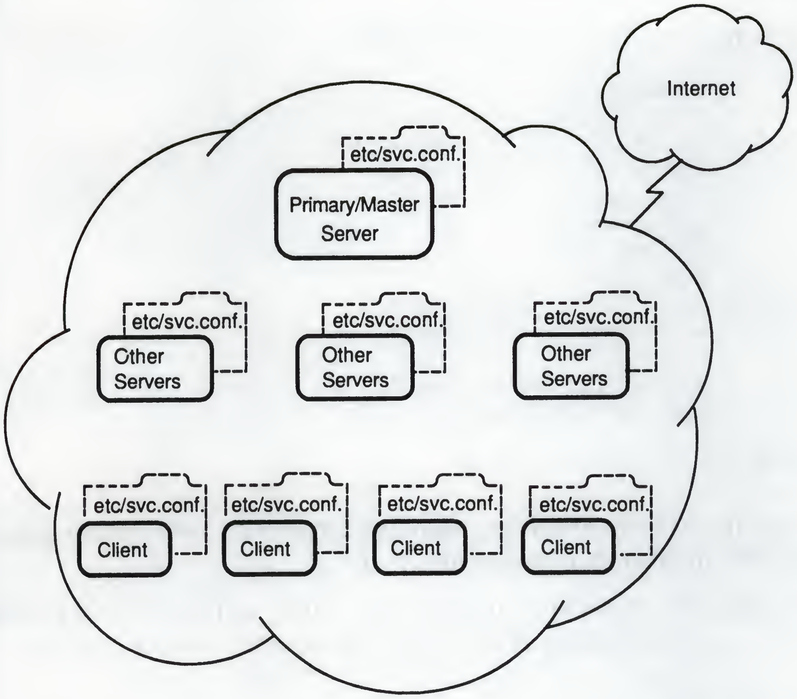
  If your network is heterogeneous and you want to distribute databases in addition to the hosts database, you must use Yellow Pages (YP). If you are only distributing the hosts database, you can use BIND/Hesiod or YP.

- For each database, the order in which the name services running on your system should be queried

  The order can be different for different databases. It is recommended that local be the first service that your system queries for all databases, regardless of what services you are running.

Steps

The following figure depicts a distributed environment that can be running BIND/Hesiod, YP, or both. You must edit the /etc/svc.conf file on each system to reflect the order in which the system should query the name services for each database that is being served.



ZK–0187U–R

You must be logged in as superuser to run the svcsetup command.

Note

> To terminate svcsetup with no modifications to the /etc/svc.conf file, press CTRL/C.

1. Type the following:

   ```
   # svcsetup
   ```

   Following some explanatory text, a configuration menu is displayed; svcsetup prompts you to select from the following options: print the current database entries (**p**), modify them (**m**), or exit the svcsetup command (**e**).

2. Select the **m** option to edit the svc.conf file:

   ```
            Configuration Menu for the /etc/svc.conf file

            Modify File      => m
            Print File       => p
            Exit             => e

   Enter your choice [m]: m
   ```

3. Select from the menu the databases whose entries you want to edit.

   The system assigns each database a number. If you are editing multiple entries, separate the database numbers by spaces.

The following example shows how to select the aliases and group database entries:

```
 Change Menu for the /etc/svc.conf file

        aliases           => 0
        auth              => 1
        group             => 2
        hosts             => 3
        netgroup          => 4
        networks          => 5
        passwd            => 6
        protocols         => 7
        rpc               => 8
        services          => 9

        all of the above  => 10
        none of the above => 11

Enter your choice(s): 0 2
```

4. Select the order in which you want the services on your system queried, and enter the number that corresponds to it.

   This example shows how to change the setting of the aliases databases to local, and the setting of the group database to local,yp:

```
        local           => 1
        yp              => 2
        bind            => 3
        local,yp        => 4
        local,bind      => 5
        yp,local        => 6
        bind,local      => 7

Enter the naming service order for the "aliases" database [5]: 1

        local           => 1
        yp              => 2
        bind            => 3
        local,yp        => 4
        local,bind      => 5
        yp,local        => 6
        bind,local      => 7

Enter the naming service order for the "group" database [5]: 4
```

After you have indicated the changes you want to make, the svcsetup command exits. The changes take effect immediately.

See Also

`svc.conf`(5), `svcsetup`(8)

Introduction to Networking and Distributed System Services
Guide to the BIND/Hesiod Service
Guide to the Yellow Pages Service

Setting Up the Network Time Services

The Network Time Protocol, alone or in combination with the Time Synchronization Protocol, allows you to synchronize time in a distributed environment.

The Network Time Protocol (NTP) provides accurate, dependable, and synchronized time for hosts on both wide area networks (like the Internet) and local area networks (LANs). In particular, NTP provides synchronization traceable to clocks of high absolute accuracy, and avoids synchronization to clocks keeping bad time. NTP is implemented by the University of Maryland's `ntpd` daemon. The `/etc/ntp.conf` file is the configuration file for the daemon.

The University of California's Time Synchronization Protocol (TSP) synchronizes workstation time. TSP is implemented by the `timed` daemon. If you use the `timed` daemon with the appropriate options, you can suppress the TSP averaging algorithm in favor of distributing NTP time. This allows you to use the `timed` daemon to distribute NTP time to machines that can not run NTP.

Before You Start

Your network must be up and running before you attempt to set up the network time services.

Before setting up the network time services, you should gather the following information:

- Whether your site is connected to the Internet or you are using a local reference clock

  If your site is connected to the Internet your best available time source is the Internet NTP service, which consists of a set of hosts on the Internet that receive time from a highly accurate source, such as a radio receiver tuned to a time code signal broadcast by a government agency. For information on finding these hosts and obtaining permission to designate them as peers, see the section of this task on setting up a primary server.

  If your site is not connected to the Internet, select a system that is wristwatch- or radio clock-controlled to be the local reference clock. The local reference clock should be your most accurate and highly available system because the time set by the local reference clock is distributed to all hosts at your site. See the section on setting up a local reference clock in this task for more information.

- The number of hosts able to run NTP

  Hosts that are running ULTRIX Version 4.0 software, or that have explicitly loaded the NTP software from a public source, can run NTP. The number of NTP hosts is used to determine the number of NTP servers needed.

- Which LANs contain hosts unable to run NTP

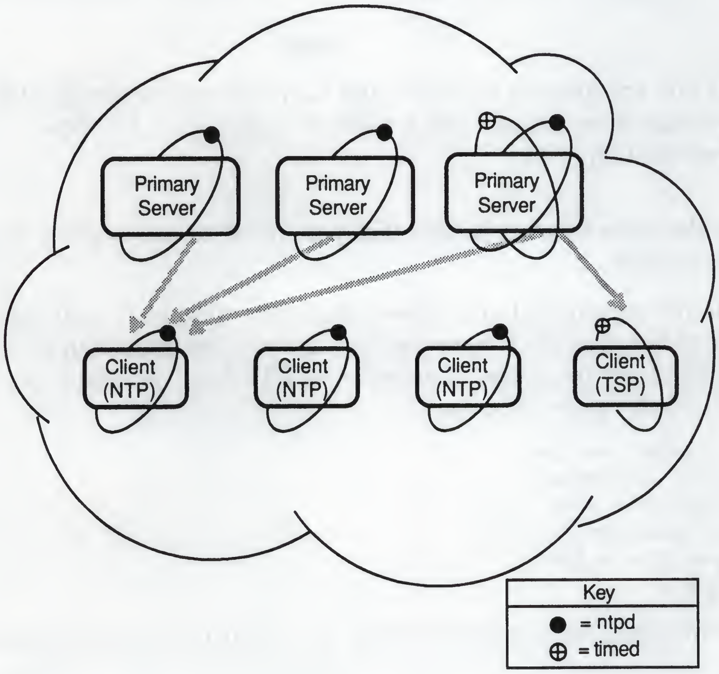
  Hosts unable to run NTP must run TSP. Each LAN with hosts running TSP must have an NTP server configured on it. The NTP server runs TSP with options that suppress TSP's averaging algorithm in favor of distributing the NTP time of the server. The number of TSP clients does not affect the number of servers.

- The Internet addresses of all servers you intend to refer to in the /etc/ntp.conf file if your site is connected to the Internet, or the local addresses of all servers if your site is not connected to the Internet

  Every host that you refer to in the /etc/ntp.conf file of a server or client must have an entry in the /etc/hosts file of that machine.

Steps

The following figure depicts a distributed environment that is running processes for time services. It illustrates the relationship between the primary servers and clients. Clients at sites where there is a local reference clock, or where fewer than 50 hosts are running NTP get their time from each of three primary servers. Clients that are unable to run NTP can get NTP time from a server that is running both NTP and TSP. The gray arrows indicate the flow of data between primary servers, a time client running NTP, and a time client running TSP.



ZK-0185U-R

If your site is connected to the Internet, complete the steps described in the section on the Internet NTP service. If your site is not connected to the Internet, complete the steps described in the section on setting up a local reference clock.

You must be logged in as superuser to set up the network time services.

Internet NTP Service – Complete the steps described in this section if your site is connected to the Internet. If your site is not connected to the Internet, go to the section on setting up a local reference clock.

Primary Server:

If your site is connected to the Internet, you should configure three NTP primary servers at your site that synchronize to three highly accurate (stratum 1 or stratum 2) hosts on the Internet.

1. Choose the three NTP primary servers.

   There should not be more than three NTP primary servers receiving time from the Internet at your site. The systems that you select should be carefully monitored and lightly loaded, if possible.

   There must be at least one server (primary or secondary) on each LAN that has clients running the `timed` daemon.

   #### Note

   > If you are running Kerberos, the Kerberos master should also be a primary time server. See the *Guide to Kerberos* for more information.

2. Choose the three Internet servers that you intend to use as peers for your primary servers.

   The Internet servers that you choose should be stratum 1 or stratum 2 servers. The list of the possible Internet servers and information about their stratum level is available by means of anonymous FTP from `louie.udel.edu`. To obtain the list, do the following:

   ```
   % ftp louie.udel.edu
   username: anonymous
   Password: your_name
   ftp> cd pub/ntp
   ftp> get clock.txt
   ftp> bye
   ```

   For security reasons, not all machines at a site may have anonymous FTP access.

   #### Note

   > Obtain permission from the contact person listed for the Internet server before specifying it as a peer.

3. For each primary server, edit the `/etc/ntp.conf` file with the host names of the Internet time servers with which your primary time servers will synchronize.

   Remove the comment character (#) from in front of each `peer` entry for which you are specifying an Internet server. Replace the {`server`*n*} entry with the host name of the Internet server.

   The following example shows a portion of a primary time server's `/etc/ntp.conf` file. The Internet time servers that the primary time server has specified as peers are `InetServer1`, `InetServer2`, and `InetServer3`. Remember that each Internet server that you specify as a peer must have an entry in the `/etc/hosts` file of the primary server.

```
#    ** S E R V E R **
#
#  If you are configuring a server, use "peer" entries to
#  synchronize to other NTP servers.  For example, server1,
#  server2, and server3.
#
peer            InetServer1
peer            InetServer2
peer            InetServer3
#
```

4. Edit the `/etc/rc.local` file.

   The `/etc/rc.local` file runs the commands located in it each time you reboot your system. You must start the `routed` daemon and place the `rdate`, `ntp`, `ntpd`, and (optionally) `timed` entries after the `syslog` entry.

   To run `routed`, remove the comment characters (#) from in front of the following lines:

   ```
   [ -f /etc/routed ] && {
          /etc/routed & echo 'routed'                  >/dev/console
   }
   ```

   Then add entries for `rdate`, `ntp`, `ntpd`, and optionally `timed`. The edited file should look similar to the following:

   ```
   [-f /etc/syslog] && {
        /etc/syslog   & echo -n ' syslog'  >/dev/console
   }
   [-f /etc/rdate] && {
        /etc/rdate -s   & echo -n ' rdate'         >/dev/console
   }
   [-f /usr/etc/ntp] && {
        /usr/etc/ntp -s -f InetServer1 InetServer2 InetServer3\
                         & echo -n ' ntp'           >/dev/console
   }
   [-f /usr/etc/ntpd] && {
        /usr/etc/ntpd -n & echo -n ' ntpd'         >/dev/console

   }
   [-f /usr/etc/timed] && {
        /usr/etc/timed -E -M & echo -n ' timed'    >/dev/console
   }
   ```

   The `/etc/rdate -s` command sets this host’s time to the approximate network time. The `/etc/rdate -s` command is included as a backup, in case all three NTP Internet servers are down when this system reboots.

   The `/usr/etc/ntp -s -f` command causes NTP to poll, in order, each of the Internet time servers specified for the time, and then synchronizes the time on this system to match that of the first Internet server to respond.

   The `/usr/etc/ntpd -n` command starts the `ntpd` daemon. The `-n` option prevents the `ntpd` program from being swapped from memory.

   The `/usr/etc/timed -E -M` command starts the `timed` daemon. You should run the `timed` daemon on your primary NTP server if your LAN has any clients that are running TSP. The `-E` option tells `timed` to distribute the time of the local machine, rather than using the TSP averaging algorithm. The `-M` option tells `timed` that this system is a time server and that it is capable of distributing time to `timed` clients. With these options set, the `timed` daemon on the server distributes NTP time to TSP clients on the LAN.

5. Reboot your system.

   Use the shutdown command with the -r option to reboot. The following command immediately performs an orderly shutdown and automatic reboot:

   ```
   # /etc/shutdown -r now
   ```

6. Verify that NTP is working correctly.

   Use the ntpdc command to verify that NTP is working correctly on your system. For information on monitoring the ntpd daemon and using the ntpdc command, see the ntpdc(8) reference page.

Secondary Server:

If your site has fewer than 50 hosts running NTP, you do not need to set up any secondary time servers. If your site has 50 or more hosts running NTP, you must configure secondary time servers.

1. Choose three NTP secondary servers for each set of up to 50 hosts running NTP.

   The systems that you select to be secondary time servers should be carefully monitored and lightly loaded, if possible.

   There must be at least one server (primary or secondary) on each LAN that has clients running the timed daemon.

2. Edit the /etc/ntp.conf file of the secondary servers with the names of the primary servers.

   You should specify the three primary time servers at your site as peers for each secondary server.

   To edit the /etc/ntp.conf file, remove the comment character (#) from in front of each peer entry. Replace the {server*n*} entry with the host name of a primary server.

   The following example shows a portion of a secondary time server's /etc/ntp.conf file. The primary time servers that the secondary time server has specified as peers are PriServer1 PriServer2 and PriServer3 Remember that each primary server that you specify as a peer must have an entry in the /etc/hosts file of the secondary server.

   ```
   #     **  S E R V E R  **
   #
   #  If you are configuring a server, use "peer" entries to
   #  synchronize to other NTP servers.  For example, server1,
   #  server2, and server3.
   #
   peer            PriServer1
   peer            PriServer2
   peer            PriServer3
   #
   ```

3. Edit the /etc/rc.local file.

   Place entries for the rdate, ntp, ntpd, and (optionally) timed commands after the syslog entry. The edited file should look similar to the following:

```
[-f /etc/syslog] && {
     /etc/syslog   & echo -n ' syslog'  >/dev/console
}
[-f /etc/rdate] && {
     /etc/rdate -s   & echo -n ' rdate'          >/dev/console
}
[-f /usr/etc/ntp] && {
     /usr/etc/ntp -s -f PriServer1 PriServer2 PriServer3\
                  & echo -n ' ntp'               >/dev/console
}
[-f /usr/etc/ntpd] && {
     /usr/etc/ntpd -n & echo -n ' ntpd'          >/dev/console

}
[-f /usr/etc/timed] && {
     /usr/etc/timed -E -M & echo -n ' timed'    >/dev/console
}
```

4. Reboot your system.

   Use the `shutdown` command with the `-r` option to reboot. The following command immediately performs an orderly shutdown and automatic reboot:

   ```
   # /etc/shutdown -r now
   ```

5. Verify that NTP is working correctly.

   Use the `ntpdc` command to verify that NTP is working correctly on your system. For information on monitoring the `ntpd` daemon and using the `ntpdc` command, see the `ntpdc`(8) reference page.

NTP Client:

Hosts that are running ULTRIX Version 4.0 software, or that have explicitly loaded the NTP software from a public source, should run NTP. Others must run TSP. For information on setting up a client to run TSP, go to the section TSP Client.

For clients that are running NTP, complete the following steps:

1. Edit the `/etc/ntp.conf` file.

   For NTP clients, remove the comment character (#) from in front of each `server` entry and replace the {`server`*n*} entries with the host names of three NTP time servers. If your site has fewer than 50 hosts running NTP, then specify the host names of the three primary time servers. If your LAN has more than 50 hosts running NTP, specify the host names of any distinct set of three secondary time servers for each set of up to 50 NTP clients.

   Note

   > Up to 50 clients can specify the same set of time servers in their `/etc/ntp.conf` file. Therefore, the network administrator may want to distribute a common `/etc/ntp.conf` file to 50 client systems.

   The following example shows a portion of a client's `/etc/ntp.conf` file at a site where there are more than 50 NTP clients. The time servers that the client is specifying are `SecServer1`, `SecServer2`, and `SecServer3`. Remember that each server that you specify must have an entry in the `/etc/hosts` file of the client system.

```
#    ** C L I E N T **
#
#  If you are configuring a client, use "server" entries to
#  synchronize to NTP servers.  For example, server1, server2,
#  and server3.
#
server          SecServer1
server          SecServer2
server          SecServer3
```

2. Edit the `/etc/rc.local` file.

   Place the `rdate`, `ntp`, and `ntpd` entries after the `syslog` entry. The edited file should look like the following:

   ```
   [-f /etc/syslog] && {
        /etc/syslog   & echo -n ' syslog'  >/dev/console
   }
   [-f /etc/rdate] && {
        /etc/rdate -s   & echo -n ' rdate'          >/dev/console
   }
   [-f /usr/etc/ntp] && {
        /usr/etc/ntp -s -f SecServer1 SecServer2 SecServer3\
                          & echo -n ' ntp'          >/dev/console
   }
   [ -f /usr/etc/ntpd ] && {
   /usr/etc/ntpd -n & echo -n ' ntpd'               >/dev/console
   }
   ```

3. Reboot your system.

   Use the `shutdown` command with the `-r` option to reboot. The following command immediately performs an orderly shutdown and automatic reboot:

   ```
   # /etc/shutdown -r now
   ```

4. Verify that NTP is working correctly.

   Use the `ntpdc` command to verify that NTP is working correctly on your system. For information on monitoring the `ntpd` daemon and using the `ntpdc` command, see the `ntpdc`(8) reference page.

TSP Client:

Hosts that are not running ULTRIX Version 4.0 software, or that have not explicitly loaded the NTP software from a public source, must run TSP (the `timed` daemon) to synchronize to NTP time. For clients running TSP, complete the following steps:

1. Edit the `/etc/rc.local` file.

   Place the `timed` entry after the `syslog` entry. The edited file should look like the following:

   ```
   [-f /etc/syslog] && {
        /etc/syslog   & echo -n ' syslog'  >/dev/console
   }
   [ -f /usr/etc/timed ] && {
        /usr/etc/timed & echo -n ' timed'   >/dev/console
   }
   ```

2. Start the timed daemon.

   The following command starts the timed daemon without rebooting the system.

   ```
   # /usr/etc/timed
   ```

   Use the timedc command to verify that TSP is working correctly on your system. For information on monitoring the timed daemon and using the timedc command, see the timedc(8) reference page.

Local Reference Clock – If you do not have access to the Internet, establish one host at your site as a local reference clock. The system that you designate as the local reference clock should be the most accurate and highly available system. All other systems at your site synchronize to this system's time.

Local Reference Clock:

To set up a local reference clock, do the following:

1. Edit the /etc/ntp.conf file of the local reference clock.

   Remove the comment character (#) from in front of the local reference clock line as follows:

   ```
   #
   peer   /dev/null       LOCL    1       -5       local
   #
   ```

2. Edit the /etc/rc.local file.

   Place the ntpd and (optionally) timed entries after the syslog entry. The edited file should look similar to the following:

   ```
   [-f /etc/syslog] && {
        /etc/syslog   & echo -n ' syslog'  >/dev/console
   }
   [ -f /usr/etc/ntpd ] && {
        /usr/etc/ntpd -n & echo -n ' ntpd'  >/dev/console
   }
   [-f /usr/etc/timed] && {
        /usr/etc/timed -E -M & echo -n ' timed'    >/dev/console
   }
   ```

   The /usr/etc/ntpd -n command starts the ntpd daemon. The -n option prevents the ntpd program from being swapped from memory.

   The /usr/etc/timed -E -M command starts the timed daemon. You should run the timed daemon on your local reference clock if your LAN has any clients that are running TSP. The -E option tells timed to distribute the time of the local machine. The -M option tells timed that this system is a time server and that it is capable of distributing time to timed clients. With these options set, the timed daemon on the server distributes NTP time to TSP clients on the LAN.

3. Reboot your system.

   Use the shutdown command with the -r option to reboot. The following command immediately performs an orderly shutdown and automatic reboot:

   ```
   # /etc/shutdown -r now
   ```

4. Verify that NTP is working correctly.

   Use the `ntpdc` command to verify that NTP is working correctly on your system. For information on monitoring the `ntpd` daemon and using the `ntpdc` command, see the `ntpdc`(8) reference page.

Primary Server:

If your site is using a local reference clock, you can configure an unlimited number of primary servers. To configure a primary server, do the following:

1. Choose three primary servers for each set of up to 50 NTP clients.

   The systems that you select should be carefully monitored and lightly loaded, if possible. There must be at least one server (primary or secondary) on each LAN that has clients running the `timed` daemon.

2. Edit the `/etc/ntp.conf` file.

   The primary servers on a network with a local reference clock should specify the local reference clock as one of their peers. They should then select two other primary servers as peers. The following example shows a portion of a primary time server's `/etc/ntp.conf` file at a site where a local reference clock is configured. The time servers that the primary time server has specified as peers are `LocalRefClock`, `PriServer1` and `PriServer2`. Remember that the host name of the local reference clock and each server that you specify as a peer must have an entry in the `/etc/hosts` file of the primary server.

```
#    ** S E R V E R **
#
#  If you are configuring a server, use "peer" entries to
#  synchronize to other NTP servers.  For example, server1,
#  server2, and server3.
#
peer            LocalRefClock
peer            PriServer1
peer            PriServer2
#
```

3. Edit the `/etc/rc.local` file.

   The `/etc/rc.local` file runs the commands located in it each time you reboot your system. Place entries for the `rdate`, `ntp`, `ntpd`, and (optionally) `timed` commands after the `syslog` entry. The `/etc/rc.local` file of primary servers at a site using a local reference clock should specify the local reference clock and two other primary servers in the `ntp` entry. The edited file should look like the following:

```
[-f /etc/syslog] && {
     /etc/syslog   & echo -n ’ syslog’  >/dev/console
}
[-f /etc/rdate] && {
     /etc/rdate -s   & echo -n ’ rdate’          >/dev/console
}
[-f /usr/etc/ntp] && {
     /usr/etc/ntp -s -f LocalRefClock PriServer1 PriServer2\
                      & echo -n ’ ntp’           >/dev/console
}
[-f /usr/etc/ntpd] && {
     /usr/etc/ntpd -n & echo -n ’ ntpd’          >/dev/console

}
[-f /usr/etc/timed] && {
     /usr/etc/timed -E -M & echo -n ’ timed’    >/dev/console
}
```

The `/etc/rdate -s` command sets this host’s time to the approximate network time. The `/etc/rdate -s` command is included as a backup, in case all three of the peer servers are down when this system reboots.

The `/usr/etc/ntp -s -f` command causes NTP to poll one of the peer servers specified for the time, and then synchronizes the time on this system to match that of the peer server.

The `/usr/etc/ntpd -n` command starts the `ntpd` daemon. The `-n` option prevents the `ntpd` program from being swapped from memory.

The `/usr/etc/timed -E -M` command starts the `timed` daemon. You should run the `timed` daemon on your primary NTP server if your LAN has any clients that are running TSP. The `-E` option tells `timed` to distribute the time of the local machine. The `-M` option tells `timed` that this system is a time server and that it is capable of distributing time to `timed` clients. With these options set, the `timed` daemon on the server distributes NTP time to TSP clients on the LAN.

4. Reboot your system.

   Use the `shutdown` command with the `-r` option to reboot. The following command immediately performs an orderly shutdown and automatic reboot:

   ```
   # /etc/shutdown -r now
   ```

5. Verify that NTP is working correctly.

   Use the `ntpdc` command to verify that NTP is working correctly on your system. For information on monitoring the `ntpd` daemon and using the `ntpdc` command, see the `ntpdc`(8) reference page.

NTP Client:

Hosts that are running ULTRIX Version 4.0 software, or that have explicitly loaded the NTP software from a public source, should run NTP. Others must run TSP. For information on setting up a client to run TSP, go to the section TSP Client.

For NTP clients, complete the following steps:

1. Edit the /etc/ntp.conf file.

   Remove the comment character (#) from in front of each server entry and replace the {server*n*} entries with the host names of three primary servers.

   The following example shows a portion of a client's /etc/ntp.conf file at a site where a local reference clock is configured. The time servers that the client is specifying are PriServer1, PriServer2, and PriServer3. Remember that each server that you specify must have an entry in the /etc/hosts file of the client system.

   ```
   #    ** C L I E N T **
   #
   #  If you are configuring a client, use "server" entries to
   #  synchronize to NTP servers.  For example, server1, server2,
   #  and server3.
   #
   server          PriServer1
   server          PriServer2
   server          PriServer3
   ```

2. Edit the /etc/rc.local file.

   Place the rdate, ntp, and ntpd entries after the syslog entry. The edited file should look like the following:

   ```
   [-f /etc/syslog] && {
        /etc/syslog   & echo -n ' syslog'   >/dev/console
   }
   [-f /etc/rdate] && {
        /etc/rdate -s   & echo -n ' rdate'           >/dev/console
   }
   [-f /usr/etc/ntp] && {
        /usr/etc/ntp -s -f PriServer1 PriServer2 PriServer3\
                          & echo -n ' ntp'           >/dev/console
   }
   [ -f /usr/etc/ntpd ] && {
   /usr/etc/ntpd -n & echo -n ' ntpd'                >/dev/console
   }
   ```

3. Reboot your system.

   Use the shutdown command with the -r option to reboot. The following command immediately performs an orderly shutdown and automatic reboot:

   ```
   # /etc/shutdown -r now
   ```

4. Verify that NTP is working correctly.

   Use the ntpdc command to verify that NTP is working correctly on your system. For information on monitoring the ntpd daemon and using the ntpdc command, see the ntpdc(8) reference page.

TSP Client:

Hosts that are not running ULTRIX Version 4.0 software, or that have not explicitly loaded the NTP software from a public source, must run TSP (the `timed` daemon) to synchronize to NTP time. For clients running TSP, complete the following steps:

1. Edit the `/etc/rc.local` file.

   Place the `timed` entry after the `syslog` entry. The edited file should look like the following:

   ```
   [-f /etc/syslog] && {
        /etc/syslog   & echo -n ’ syslog’  >/dev/console
   }
   [ -f /usr/etc/timed ] && {
        /usr/etc/timed & echo -n ’ timed’   >/dev/console
   }
   ```

2. Start the `timed` daemon.

   The following command starts the `timed` daemon without rebooting the system.

   ```
   # /usr/etc/timed
   ```

   Use the `timedc` command to verify that TSP is working correctly on your system. For information on monitoring the `timed` daemon and using the `timedc` command, see the `timedc`(8) reference page.

See Also

`ntp`(1), `ntp.conf`(5), `ntpd`(8), `ntpdc`(8), `timed`(8), `timedc`(8)

Introduction to Networking and Distributed System Services
Guide to Kerberos
RFC 1129—*Internet time synchronization: the Network Time Protocol*

Adding Users in a Distributed Environment

Adding user accounts in a distributed environments differs from adding user accounts to a single machine. To add user accounts if your LAN is distributing the `passwd` database with either Yellow Pages (YP) or BIND/Hesiod, you must edit and propagate the database from the master server manually.

Note

> If the master server for the `passwd` database is running at the UPGRADE or ENHANCED security level, you must use BIND/Hesiod to distribute the `passwd` and `auth` databases. Adding user accounts in an UPGRADE or ENHANCED environment is described in the *Security Guide for Administrators.*

Before You Start

Before editing the `passwd` database with a new user account, you should gather the following information:

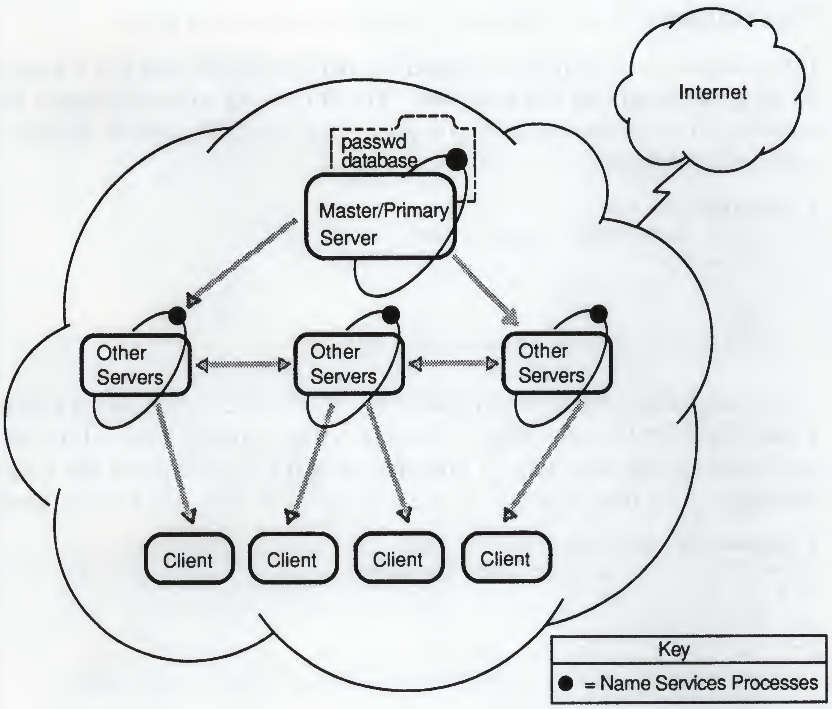
- Which naming service your LAN is using to distribute the `passwd` database

  The `passwd` database is located in the `/var/yp/src` directory if you are using YP, and in the `/var/dss/namedb/src` directory if you are using BIND/Hesiod.

- The following information on the new user:
  - Login name
  - User identification number (UID)
  - Group identification number (GID)
  - Real name, office number, and telephone extension
  - Initial working directory
  - Program to use as a shell

Steps

The following figure depicts a distributed environment in which the `passwd` database is being served by either BIND/Hesiod or YP. Any changes to the database are made on the master or primary server; updates are distributed to the other servers. Clients query a server for password information. The gray arrows indicate the flow of data between the master or primary server, other servers, and clients.



ZK-0186U-R

You must be logged in as superuser to complete the following steps:

1. Change to the directory where the passwd database is located.

   If you are distributing the passwd database with YP, it is located in /var/yp/src; if you are distributing the passwd database with BIND/Hesiod, it is located in /var/dss/namedb/src.

2. Edit the passwd database with an entry for the new user.

   The format for each user account is the same as the format in the /etc/passwd file:

   login-name:password field:UID:GID:user-info:initial-working-directory:shell-program

   Set the *password* field to Nologin.

3. Rebuild the password database.

   If the database is being distributed by BIND/Hesiod, change to the /var/dss/namedb directory and run the make passwd command. The following sequence of commands shows you how to rebuild the passwd database:

   ```
   # cd /var/dss/namedb
   # make passwd
   ```

   If the database is being distributed by YP, change to the /var/yp directory and run the make passwd command. The following sequence of commands shows you how to rebuild the passwd database:

   ```
   # cd /var/yp
   # make passwd
   ```

4. Set a password for the new user.

   Note that the new user can not log in if no password is set.

   If the database is being distributed by BIND/Hesiod, use the `passwd` command to set a password for the new user. The following example shows how to set a password for the new user if you are using BIND/Hesiod to distribute the `passwd` database:

   ```
   # passwd new_user
   Changing password for new_user
   Old password:
   Enter new password:
   Verify:
   Your distributed password is updated
   ```

   If the database is being distributed by YP, use the `yppasswd` command to set a password for the new user. The following example shows how to set a password for the new user if you are using YP to distribute the `passwd` database. The host `host1.cities.dec.com` is the YP master server.

   ```
   # yppasswd new_user
   Changing yp password for new_user
   Old yp password:
   New password:
   Retype new password:
   yellow pages passwd changed on host1.cities.dec.com
   ```

   Note

   If you are sharing the `/etc` source files for BIND/Hesiod and YP by using symbolic links, beware of changes to the `passwd` database. The `passwd` and `yppasswd` commands run independently and do not use a lock mechanism on the file. In simultaneous updates of the `passwd` database, this could result in the loss of one of the updates.

See Also

`passwd(1)`, `yppasswd(1)`

Introduction to Networking and Distributed System Services
Guide to the BIND/Hesiod Service
Guide to the Yellow Pages Service

Index

How to Order Additional Documentation

Technical Support

If you need help deciding which documentation best meets your needs, call 800-343-4040 before placing your electronic, telephone, or direct mail order.

Electronic Orders

To place an order at the Electronic Store, dial 800-234-1998 using a 1200- or 2400-baud modem from anywhere in the USA, Canada, or Puerto Rico. If you need assistance using the Electronic Store, call 800-DIGITAL (800-344-4825).

Telephone and Direct Mail Orders

Your Location	Call	Contact
Continental USA, Alaska, or Hawaii	800-DIGITAL	Digital Equipment Corporation P.O. Box CS2008 Nashua, New Hampshire 03061
Puerto Rico	809-754-7575	Local Digital Subsidiary
Canada	800-267-6215	Digital Equipment of Canada Attn: DECdirect Operations KAO2/2 P.O. Box 13000 100 Herzberg Road Kanata, Ontario, Canada K2K 2A6
International	————	Local Digital subsidiary or approved distributor
Internal*	————	SSB Order Processing - WMO/E15 *or* Software Supply Business Digital Equipment Corporation Westminster, Massachusetts 01473

* For internal orders, you must submit an Internal Software Order Form (EN-01740-07).

Reader's Comments

ULTRIX
Guide to System and Network Setup
AA-ME88B-TE

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

Please rate this manual:	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	□	□	□	□
Completeness (enough information)	□	□	□	□
Clarity (easy to understand)	□	□	□	□
Organization (structure of subject matter)	□	□	□	□
Figures (useful)	□	□	□	□
Examples (useful)	□	□	□	□
Index (ability to find topic)	□	□	□	□
Page layout (easy to find information)	□	□	□	□

What would you like to see more/less of? ____________________

What do you like best about this manual? ____________________

What do you like least about this manual? ____________________

Please list errors you have found in this manual:

Page ______ Description ____________________

Additional comments or suggestions to improve this manual:

What version of the software described by this manual are you using? __________

Name/Title ____________________ Dept. __________

Company ____________________ Date __________

Mailing Address ____________________

__________ Email __________ Phone __________

Do Not Tear – Fold Here and Tape

digital™

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST-CLASS MAIL PERMIT NO. 33 MAYNARD MA

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION
OPEN SOFTWARE PUBLICATIONS MANAGER
ZKO3-2/Z04
110 SPIT BROOK ROAD
NASHUA NH 03062-9987

Do Not Tear – Fold Here

Cut
Along
Dotted
Line

Reader's Comments

ULTRIX
Guide to System and Network Setup
AA-ME88B-TE

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

Please rate this manual:	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	□	□	□	□
Completeness (enough information)	□	□	□	□
Clarity (easy to understand)	□	□	□	□
Organization (structure of subject matter)	□	□	□	□
Figures (useful)	□	□	□	□
Examples (useful)	□	□	□	□
Index (ability to find topic)	□	□	□	□
Page layout (easy to find information)	□	□	□	□

What would you like to see more/less of? ____________________

____

____

What do you like best about this manual? ____________________

____

____

What do you like least about this manual? ____________________

____

____

Please list errors you have found in this manual:

Page ______ Description ____________________

______ ____________________

______ ____________________

______ ____________________

______ ____________________

Additional comments or suggestions to improve this manual:

____

____

____

____

What version of the software described by this manual are you using? ______

Name/Title ____________________ Dept. ______

Company ____________________ Date ______

Mailing Address ____________________

______ Email ______ Phone ______

Do Not Tear – Fold Here and Tape

digital™

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

DIGITAL EQUIPMENT CORPORATION
OPEN SOFTWARE PUBLICATIONS MANAGER
ZKO3–2/Z04
110 SPIT BROOK ROAD
NASHUA NH 03062–9987

Do Not Tear – Fold Here

Cut
Along
Dotted
Line

digital